STATISTICS, COMPUTATION & APPLICATIONS

A Dissertation
Presented to
The Academic Faculty

By

Chuanping Yu

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Industrial and Systems Engineering

Georgia Institute of Technology

May 2020

STATISTICS, COMPUTATION & APPLICATIONS

Approved by:

Dr. Xiaoming Huo, Advisor
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Valerie Thomas
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Jianjun Shi
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Yajun Mei
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Yao Xie
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Wenjing Liao
School of Mathematics
Georgia Institute of Technology

Date Approved: March 25, 2020

To my parents

ACKNOWLEDGEMENTS

This dissertation would not have been possible without the support of so many people during my five-year PhD study. I would like to express my sincere appreciation to everyone who has given me any kind of help.

First, I am deeply indebted to my advisor, Prof. Xiaoming Huo for his endless support and guidance, and his immense knowledge and penetrating insights. He is not only a top researcher but also a good advisor who trusts me and encourages me to explore and pursue in research. I have learned a lot from him not only in the academic aspect but also in the life attitudes. I feel tremendously lucky and grateful for having him as my advisor.

I would also like to thank the rest of my dissertation committee: Prof. Valerie Thomas, Prof. Jianjun Shi, Prof. Yajun Mei, Prof. Yao Xie, and Prof. Wenjing Liao, for their invaluable advice on my research and insightful guidance on my life. It is my honor to be encouraged and guided by them and I truly appreciate it.

My sincere thanks also go to Dr. Xuezhou Mao, Dr. Na An, Dr. Chyi-Fu Hong, and all the colleagues I have worked with in Sanofi and Amazon, for their great support during my summer internships. Their insights of solving real world problems open my mind and eventually prompt me to go on the career path of industrial research.

I am also thankful to my mates and colleagues at Georgia Tech. I will never forget their company through all my ups and downs. I would also like to thank my friends from USTC - my undergraduate university in China. Although they are now in different places around the world after graduation, their support has never left me.

Last but not the least, I would like to express my gratitude to my parents. None of this would have become possible without the love and support from them.

TABLE OF CONTENTS

Acknowledgments iv

List of Tables x

List of Figures xi

Chapter 1: Distance-Based Independence Screening for Canonical Analysis . . . 1

1.1 Introduction 1

1.2 Methodology 5

1.2.1 Distance Covariance 5

1.2.2 Problem Formulation 6

1.2.3 Motivation 7

1.2.4 Independence Test 8

1.2.5 DISCA algorithm 9

1.2.6 Estimating u^* 11

1.3 Theoretical Results 12

1.3.1 Consistency Properties 12

1.3.2 Comparison with Existing Methods 13

1.4 Applying DCA for estimating u^* 15

1.4.1 Review of DCA 16

1.4.2 Minimizing $\mathcal{V}_N^2(\mathbf{X}u, \mathbf{Y})$ 16

1.4.3 Solving the Subproblem 19

1.4.4 Convergence Analysis 21

1.5 Simulation Studies 21

1.5.1 Counterexample Simulation 22

1.5.2 Comparison with Existing Methods 23

1.5.3 LA Pollution-Mortality Study (1970-1979) 28

1.6 Conclusion 29

Chapter 2: Optimal Projections in the Distance-Based Statistical Methods . . . 31

2.1 Introduction 31

2.2 Problem formulation 35

2.3 Derivable analytical results 37

2.3.1 Special case when the dimension is 2 37

2.3.2 Second special case with provable result 39

2.4 Numerical approach in general cases 42

2.5 Simulations 45

2.5.1 When the dimension is 2 46

2.5.2 When we have $n = p$ 46

2.5.3 General setting: $n \geq p$ 47

2.6 Conclusion 50

Chapter 3: A New Semidefinite Programming Algorithm for Power Flow and Power System State Estimation 51

3.1 Introduction 51
3.1.1 Previous Studies 52
3.1.2 Our Contributions 54
3.1.3 Notations 55
3.2 Preliminaries 55
3.3 Power Flow Analysis 57
3.4 Power System State Estimation 60
3.5 Convergence Analysis 62
3.6 Numerical Tests 63
3.6.1 Power Flow (PF) Simulation Results 64
3.6.2 Power System State Estimation (PSSE) Simulation Results 66
3.7 Conclusion 73

Appendix A: Proofs in Chapter 1 75

A.1 Proof of Lemma 1.2.8 75
A.2 Proof of Lemma 1.2.11 75
A.3 Proof of Lemma 1.3.1 78
A.4 Proof of Theorem 1.3.3 78
A.5 Proof of Theorem 1.3.4 79
A.6 Proof of Lemma 1.4.2 79
A.7 Proof of Proposition B.3.2 80
A.8 Proof of Proposition B.3.1 80
A.9 Proof of Lemma 1.4.3 81

A.10 Proof of Lemma 1.4.4 82

A.11 Proof of Theorem 1.4.5 84

Appendix B: Proofs in Chapter 2 87

B.1 Proof of Theorem 2.2.1 87

B.2 Proof of Theorem 2.3.1 88

B.3 Propositions we need in order to prove Theorem 2.3.2 94

B.4 Proof of Theorem 2.3.2 98

B.5 Proof of Theorem 2.3.3 104

B.6 Proof of Lemma 2.3.4 106

B.7 Proof of Lemma 2.3.5 107

B.8 Proof of Lemma 2.3.7 108

B.9 Proof of Theorem 2.3.8 109

B.10 Proof of Lemma 2.4.1 113

B.11 Proof of Theorem 2.4.2 115

Appendix C: Proofs in Chapter 3 118

C.1 Proof of Proposition 3.3.1 118

C.2 Proof of Theorem 3.3.2 119

C.3 Proof of Theorem 3.3.3 120

C.4 Proof of Theorem 3.5.1 120

C.5 Proof of Theorem 3.5.3 121

C.6 Convergence Analysis for State Estimation 123

C.6.1 Proof of Theorem 3.5.2 123

References 130

LIST OF TABLES

1.1 Table of the dimension of $\widehat{W}_X$ 23

1.2 Summary of the LA Pollution-Mortality Data 28

1.3 DISCA reduced the 8-dimensional space of X into a 3-dimensional subspace, with basis vectors shown as the rows in the above table. 29

LIST OF FIGURES

1.1 Boxplot of $dist(\widehat{W}_X, W_X)$ 23

1.2 The above figures are the results of Example 1: figures on the first row are the boxplots of $dist(\widehat{W}_X, W_X)$ obtained by DISCA and DCS respectively; figures on the bottom row are the boxplots of $dist(\widehat{W}_Y, W_Y)$ obtained by DISCA and DCS respectively. The x-axis represents $N = 50, 100, 150, 200$. 25

1.3 The above figures are the results of Example 2: the figures in the first row are the boxplots of $dist(\widehat{W}_X, W_X)$ obtained by DISCA, CCA, and DCS respectively; the figures in the bottom row are the boxplots of $dist(\widehat{W}_Y, W_Y)$ obtained by DISCA, CCA, and DCS respectively. The x-axis represents $N = 50, 100, 150, 200$. 26

1.4 The above figures are the results of Example 3: the figures in the first row are the boxplots of $dist(\widehat{W}_X, W_X)$ obtained by DISCA, CCA, and DCS respectively; the figures in the second row are the boxplots of $dist(\widehat{W}_Y, W_Y)$ obtained by DISCA, CCA, and DCS respectively. The x-axis represents $N = 50, 100, 150, 200$. 27

2.1 Optimal projection vs. Monte Carlo in the 2 dimensional case 47

2.2 Optimal projection vs. Monte Carlo in the $n = p$ case 48

2.3 Optimal projection vs. Monte Carlo for dimension varying from 3 to 7 in the case $n = 8$ 48

2.4 Optimal projection vs. Monte Carlo for dimension varying from 3 to 8 in the case $n = 9$ 49

2.5 Optimal projection vs. Monte Carlo for dimension varying from 3 to 9 in the case $n = 10$ 49

2.6 Optimal projection vs. Monte Carlo for dimension varying from 3 to 10 in the case $n = 11$ 50

3.1 9-bus system power flow problem simulation 65

3.2 30-bus system power flow problem simulation 65

3.3 57-bus system power flow problem simulation 65

3.4 Our method vs. Zhang's method vs. WLS in the 57-bus system. 66

3.5 Our method vs. Zhang's method vs. WLS in the 118-bus system. 67

3.6 Simulation results with noises in the 9-bus system. 67

3.7 Simulation results with $N(0, 0.001^2)$ and $N(0, 0.01^2)$ noises in the 57-bus system. 68

3.8 Simulation results with $N(0, 0.001^2)$ noises in the 118-bus system. 69

3.9 Bad data simulation results with $N(0, 0.001^2)$ noises in the 9-bus system. . 70

3.10 Bad data simulation results with $N(0, 0.001^2)$ noises in the 57-bus system. . 71

3.11 Ten errors with $N(0, 0.001^2)$ noises in the 118-bus system. 72

3.12 Different start values with $N(0, 0.001^2)$ noises in the 57-bus system. 72

SUMMARY

When statistics meets real applications, the computational aspect of the statistical methods becomes critical. In this thesis, I improve the computational efficiency of some statistical methods, so that they become both computationally and statistically optimal. Inspired by the recent development of the distance-based methods in statistics, I first propose a novel distance-based canonical analysis method for effective dimension reduction. Secondly, an efficient algorithm of calculating distance-based statistics is studied. Moreover, a new semidefinite programming algorithm is also developed for the applications in power flow analysis problems; it appears to be more robust than existing methods.

I give more details in the following. In the first part of this dissertation, we introduce a novel dimension reduction method called distance-based independence screening for canonical analysis (DISCA), which can be used to reduce dimensions of two random vectors with arbitrary dimensions. The essence of our method – DISCA – is to use the distance-based independence measure – distance correlation, which was proposed by Székely and Rizzo in 2007 – to eliminate the 'redundant' dimensions until infeasible. Numerically, DISCA is to solve a non-convex optimization problem. Algorithms and theoretical justifications are provided, and the comparisons with other existing methods demonstrate its accuracy, universality, and effectiveness. An R package *DISCA* can be found on GitHub.

Noticing that distance correlation used in DISCA is computationally expensive with the increase of space dimensions, in the second part of this dissertation, we manage to accelerate the calculation of distance-based statistics, by projecting multidimensional variables onto pre-specified projection directions, with the improvement of computational complexity from $O(m^2)$ to $O(nm \cdot \log(m))$, where n is the number of projection directions and m is the sample size. Computational savings are achieved when $n \ll m/\log(m)$. The optimal pre-specified projection directions can be obtained by minimizing the worse-case difference between the true distance and the approximated distance. We provide solutions

and greedy algorithms for different scenarios, and confirm the advantage of our technique in comparison with the pure Monte Carlo approach, in which the directions are randomly selected rather than pre-calculated.

In the third part of this dissertation, we turn our focus on the applications of statistical computational algorithms in power systems area. A new semidefinite programming algorithm is proposed to solve the power flow and power system state estimation problems. Both two kinds of problems are non-convex, and convex relaxation is the typical approach to non-convexity in power systems area, while the objective functions are required to be carefully designed in order to keep the equivalency before and after relaxation. We first reformulate the two types of complex-valued problems as non-convex real-valued ones. We show that an alternating semidefinite programming algorithm can be applied and is not sensitive to the start point without the sacrifices of accuracy. Furthermore, it performs well even when the voltage angles are not close to zero. Convergence analysis is provided, and numerical studies on representative power systems datasets demonstrate the accuracy of our proposed algorithm, and applicability on various scenarios of different given measurements.

CHAPTER 1
DISTANCE-BASED INDEPENDENCE SCREENING FOR CANONICAL ANALYSIS

1.1 Introduction

The problem that this chapter focuses on, is to peel off the "redundant" dimensions between two random vectors such that any further dimension reduction by linear projections will lose the dependency structure (linear or nonlinear) between the two random vectors. In this chapter, we propose a new backward eliminating method, called distance-based independence screening for canonical analysis (DISCA), based on the distance covariance to carry out dimension reduction for two sets of random vectors. Distance covariance, proposed by [20], is a measure of dependence between two arbitrarily-dimensional random vectors. It can be used to perform the independence testing for both continuous and discrete distributions, and to detect both linear and nonlinear relationships. Our distance-covariance-based strategy is to utilize distance covariance as a criterion to remove the independent structures until further elimination would bring the loss of dependency information between the two random vectors. DISCA does not require any distributional assumption or any data structure assumption (such as the additional assumption in [23], and [22]). It can handle both equal and unequal dimension reduction cases. Moreover, it can confirm the effective subspaces as well as their dimensions simultaneously and does not require other sub-sampling techniques (such as the bootstrap) to estimate the dimensions of the subspaces at the beginning.

Our problem can be roughly seen as a canonical correlation analysis (CCA) problem. Ever since [4] proposed the canonical correlation analysis (CCA), to extend the classical CCA to the nonlinear (non-Gaussian) cases, many methods have been introduced, such as

Kernel CCA by [7, 8] and [9], Informational CCA by [14], deep CCA by [6], HSIC-CCA by [10], etc. DISCA is an improvement of all the CCA methods in the sense that, first, it can detect not only equal dimensional dependent structure, which are the pairs of canonical variables, but also non-equal dimensional dependent structure; second, it does not need appropriately chosen kernel functions or nonlinear model functions as in Kernel CCA, HSIC-CCA, and deep CCA; third, DISCA does not involve density estimation, which is a difficult problem and computationally expensive, as in Informational CCA. Besides the above improvements, DISCA still keeps the advantages in the performance when non-normality and nonlinear relationships exist.

The dimension reduction problem, from the regression viewpoint, can also be roughly viewed as a Sufficient Dimension Reduction (SDR) problem. The major assumption in an SDR problem is

$$Y \perp\!\!\!\perp X | \beta^T X, \tag{1.1}$$

where $\perp\!\!\!\perp$ stands for statistical independence between two random variables (or vectors), and Y is a random variable (or vector) in $\mathbb{R}^q$, X is a random vector in $\mathbb{R}^p$, $\beta \in \mathbb{R}^{p\times r}(r \leq p)$ is a matrix, and the space spanned by the columns of β is called the central subspace, denoted as $\mathcal{S}_{Y|X}$. Finding the central subspace (i.e., the β) is the main task of the SDR. The majority of SDR methods only handle the case when Y is univariate, such as the sliced inverse regression (SIR) [16], the sliced average variance estimation (SAVE) [34], the sliced regression (SR) [35], and so on. These methods have certain restrictions for X and Y, such as non-symmetric dependency, existed second moment, or sample size being larger than dimension p. Since the direct multivariate extensions of most univariate SDR methods do not work well because of the "curse of dimensionality" (taking slicing method for example, when univariate, Y can be sliced into several intervals but when Y is multivariate, due to the multiplicative nature of slicing in a multi-dimensional space, even when Y is just three dimensions, how to slice the three-dimension space into pieces can be a challenge in numerics.), projection methods are introduced to make the univariate methods applicable

to the multivariate responses by projecting Y on one-dimensional spaces and then utilizing the univariate SDR methods, such as [33] with 'optimal' projections, [37] with random projections, and so on. Because these multivariate methods still need to apply the univariate methods, the limitations existing in the univariate methods can not be avoided. Besides, since it is a regression problem, X and Y are not equally treated and not exchangeable, which makes infeasible to the problem setting in our case.

Sheng and Yin in 2013 and 2016 discuss how to find the central subspace based on the distance covariance in two different cases: (case 1) β is a vector [23], and (case 2) β is a matrix [22]. The assumption they used, however, is stronger than the original setting of SDR: in Sheng and Yin's papers, they need another assumption in addition to (1.1) to make their theorems work:

$$P_\beta^T X \perp\!\!\!\perp (I - P_\beta)^T X, \tag{1.2}$$

where P_β denotes the projection operator that projects onto the space spanned by the columns of β, and I is the identity matrix. From (1.1) and (1.2) we know that Y is independent of $(I - P_\beta)^T X$, which is equivalent to say $P_{W_X^\perp} X \perp\!\!\!\perp Y$ as in Assumption 1.2.6, but (1.1) and (1.2) cannot be derived from the latter. So we have the following proposition showing that our assumption is weaker than theirs.

Proposition 1.1.1. *Assumption 1.2.6 in DISCA is weaker than assumptions* (1.1) *and* (1.2) *in [23] and [22].*

DISCA is also an improvement of SDR methods in the way that it is a general methods for either univariate or multivariate Y, and can be used for equally or unequally treated X and Y which makes find the central subspace for Y in terms of X as a special case.

[19] improves the SDR methods in similar spirit as ours. They define the dual central subspace (DCS) in place of the previous central subspace $\mathcal{S}_{X|Y}$, and propose a method to get both the central and dual central subspaces at the same time based on minimizing the Kullback-Leibler (KL) divergence, which relies on the Gaussian kernel to estimate

density. DISCA is advantageous even compared with DCS as we do not need to do density estimation. (Note that kernel-based density estimation can be very sensitive to the curse-of-dimensionality.) Another improvement is that they use bootstrap technique, the same as in [23] and [22], to estimate the dimension of both subspaces initially, which brings computational burdens, while DISCA does not need this step.

To illustrate the effectiveness of DISCA, in the following we construct a simple example where DISCA can prevail while all the CCA, SDR, and DCS methods fail. Suppose we have two random vectors $X = (X_1, \ldots, X_p)^T \in \mathbb{R}^p, Y = (Y_1, \ldots, Y_q)^T \in \mathbb{R}^q$ satisfying

$$Y_j = f_j\left(\sum_{i=1}^{p} X_i\right) + \epsilon_j, j = 1, \ldots, q,$$

where f_j's are functions in which at least two are the same, and ϵ_j's are random noises independent of X. The true dimensional reduction subspace for X would be the one spanned by $(\frac{1}{\sqrt{p}}, \frac{1}{\sqrt{p}}, \ldots, \frac{1}{\sqrt{p}})^T$, and the dimension of the reduced subspace for Y depends on how many f_j's are the same: for example, if two of the functions are the same, the reduced subspace for Y would be $q-1$. In this simple case, we will argue (in Section 1.3.2) that the proposed DISCA can recover the fundamental dependency structure, while the other methods cannot. More explanations can be found in Section 1.5.1, which also gives a specific simulation example based on this design.

The remainder of this chapter is organized as follows. We present background material on the distance covariance and develop our methodology in Section 1.2, followed by the theoretical results in Section 1.3. Section 1.4 contains both the algorithm and the convergence analysis of the proposed algorithm. In Section 1.5, simulation examples are given for comparison with the existing methods, and showing the capability of our method in the unsolvable cases of other methods. Finally, we conclude and discuss future works in Section 2.6. When possible, all proofs are relegated to the appendix.

1.2 Methodology

In this section, we first give a brief review of the distance covariance, a measure of the independence between two random vectors in Section 1.2.1, and then describe the formulation of our problem in Section 2.2, followed by the motivation of our strategy in Section 1.2.3 and 1.2.4. In Section 1.2.5 and 1.2.6, we introduce our method in details.

1.2.1 Distance Covariance

Let X and Y be two random vectors from $\mathbb{R}^p$ and $\mathbb{R}^q$, respectively. We denote $|\cdot|_p$ as the Euclidean norm in $\mathbb{R}^p$, and $|\cdot|_q$ as the Euclidean norm in $\mathbb{R}^q$. The distance covariance (dCov) between random vectors X and Y with finite first moments is the nonnegative number $\mathcal{V}(X,Y)$ that is defined in [20], and in [46], an equivalent and more numerically amenable version is introduced as follows.

Definition 1.2.1. *(Theorem 8 in [46]) We have*

$$\begin{aligned}\mathcal{V}^2(X,Y) &= \mathbb{E}\left[|X-X'|_p|Y-Y'|_q\right] - \mathbb{E}\left[|X-X'|_p|Y-Y''|_q\right] \\ &\quad -\mathbb{E}\left[|X-X''|_p|Y-Y'|_q\right] + \mathbb{E}\left[|X-X'|_p\right]\mathbb{E}\left[|Y-Y'|_q\right],\end{aligned}$$

where $(X,Y),(X',Y'),$ *and* (X'',Y'') *are i.i.d.*

Let $(\mathbf{X},\mathbf{Y})$ be our N samples of random vector X and Y: $\mathbf{X}\in\mathbb{R}^{N\times p}$, $\mathbf{Y}\in\mathbb{R}^{N\times q}$. Each row of $(\mathbf{X},\mathbf{Y})$ represents one observation of X and Y. The empirical distance covariance can be written as follows.

Definition 1.2.2. *(Empirical distance covariance) We have*

$$\mathcal{V}_N^2(\mathbf{X},\mathbf{Y}) = S_1(\mathbf{X},\mathbf{Y}) + S_2(\mathbf{X},\mathbf{Y}) - 2S_3(\mathbf{X},\mathbf{Y}),$$

where

$$
\begin{aligned}
S_1(\mathbf{X},\mathbf{Y}) &= \frac{1}{N^2}\sum_{i,j=1}^{N}|X_i-X_j|_p|Y_i-Y_j|_q,\\
S_2(\mathbf{X},\mathbf{Y}) &= \frac{1}{N^2}\sum_{i,j=1}^{N}|X_i-X_j|_p\frac{1}{N^2}\sum_{i,j=1}^{N}|Y_i-Y_j|_q,\\
S_3(\mathbf{X},\mathbf{Y}) &= \frac{1}{N^3}\sum_{i=1}^{N}\sum_{j,m=1}^{N}|X_i-X_j|_p|Y_i-Y_m|_q.
\end{aligned}
\tag{2.3}
$$

The following are some results that are quoted from [20] and will be used in this chapter. Theorem 1.2.3 shows that the independence of two random vectors are equivalent to their distance covariance being zero. Theorem 1.2.4 describes the asymptotic property of the empirical distance covariance.

Theorem 1.2.3. *(Theorem 3 in [20]) We have*

$$\mathcal{V}^2(X,Y)=0 \text{ if and only if } X \text{ and } Y \text{ are independent.} \tag{2.4}$$

Theorem 1.2.4. *(Theorem 2 in [20]) If $\mathbb{E}|X|_p<\infty, \mathbb{E}|Y|_q<\infty$, then we have almost surely*

$$\lim_{N\to\infty}\mathcal{V}_N(\mathbf{X},\mathbf{Y})=\mathcal{V}(X,Y).$$

1.2.2 Problem Formulation

We consider two random vectors $X\in\mathbb{R}^p, Y\in\mathbb{R}^q$ satisfying the following two assumptions. Assumption 1.2.5 is a regular assumption in order to make sure that the distance covariance exists; Assumption 1.2.6 is the one and only assumption related to our model: we assume that there exists some subspace $W_X^{\perp}$ in the space of $\mathbb{R}^p$ and $W_Y^{\perp}$ in the space of $\mathbb{R}^q$, such that the projection of X on $W_X^{\perp}$ is independent of Y, and symmetrically the projection of Y on $W_Y^{\perp}$ is independent of X. The objective is to find the orthogonal complements of $W_X^{\perp}$ and $W_Y^{\perp}$, which are denoted as W_X and W_Y respectively.

Assumption 1.2.5. *We have $\mathbb{E}|X|_p < \infty, \mathbb{E}|Y|_q < \infty$, where $|\cdot|_p$ is the Euclidean norm in the R^p space, and $|\cdot|_q$ is the Euclidean norm in the R^q space.*

Assumption 1.2.6. *Assume there exists W_X, a p_0-dimensional ($p_0 \leq p$) subspace of $\mathbb{R}^p$, and W_Y, a q_0-dimensional ($q_0 \leq q$) subspace of $\mathbb{R}^q$, such that their orthogonal complement $W_X^\perp$ and $W_Y^\perp$ are the "largest"subspaces satisfying*

$$P_{W_X^\perp} X \perp\!\!\!\perp Y \quad \textit{and} \quad P_{W_Y^\perp} Y \perp\!\!\!\perp X,$$

where $P_{W_X^\perp} X$ (or $P_{W_Y^\perp} Y$, resp.) stands for the projection of vector X (or Y, resp.) to the subspace $W_X^\perp$ (or $W_Y^\perp$, resp.). The "largest"subspace means that for any $W^\perp$ that is a linear subspace of $\mathbb{R}^p$ and satisfies $P_{W^\perp} X \perp\!\!\!\perp Y$, the dimension of $W^\perp$ cannot be larger than $p - p_0$; Similarly, for any $W^\perp$ that is a linear subspace of $\mathbb{R}^q$ and satisfies $P_{W^\perp} Y \perp\!\!\!\perp X$, its dimensionality cannot be larger than $q - q_0$.

If we know the subspaces W_X and W_Y, X and Y are reduced to $P_{W_X} X$ and $P_{W_Y} Y$. Then we achieve the objective of dimension reduction for both X and Y.

1.2.3 Motivation

Our strategy of finding space W_X and W_Y is motivated by Theorem 3 in [20] (listed as Theorem 1.2.3 in this chapter): we aim to find all the directions on which the projection of X is independent of Y, which is equivalent to finding all u's ($u \in S^{p-1}$) such that $\mathcal{V}^2(u^T X, Y) = 0$, where S^{p-1} refers to the unit sphere in $\mathbb{R}^p$. Then W_X is the orthogonal complement of $W_X^\perp$, and $W_X^\perp$ is the space spanned by all the directions u we have found. In terms of the exchangeability of X and Y in Assumption 1.2.6, subspace W_Y can be found by switching the positions of X and Y in the above and use $P_{W_X} X$ instead of X. Since we have Lemma 1.2.7, which can be derived by $\mathcal{V}^2(u^T X, Y) \geq 0$, and $P_{W_X}^\perp X \perp\!\!\!\perp Y$ in Assumption 1.2.6, instead of finding all the $u \in S^{p-1}$ such that $\mathcal{V}^2(u^T X, Y) = 0$, all we need to do is to find all the directions that minimize $\mathcal{V}^2(u^T X, Y)$, which can be further

formulated as finding all the orthonormal directions that minimize $\mathcal{V}^2(u^T X, Y)$, since any linear subspace can be determined by its orthonormal basis.

Lemma 1.2.7. *If there exists some $u \in \mathbb{R}^p$ such that $\mathcal{V}^2(u^T X, Y) = 0$, then finding the direction $u \in \mathbb{R}^p$ such that $\mathcal{V}^2(u^T X, Y) = 0$ is equivalent to finding the solution of*

$$\min \left\{ \mathcal{V}^2(u^T X, Y) : u \in S^{p-1} \right\},$$

where S^{p-1} is the unit sphere in $\mathbb{R}^p$.

This inspires us to develop an iteration: finding one direction each time, and computing for the next one (if it exists) in the linear subspace that is the orthogonal complement of the subspace spanned by the obtained directions. The following lemma shows that our algorithm can help us to obtain the desired directions.

Lemma 1.2.8. *Under the Assumption 1.2.5 and 1.2.6, assume $W = span(S)$ is a subspace of $\mathbb{R}^p$ satisfying $W^\perp = span(S^\perp) \subset W_X^\perp$, where S and $S^\perp$ are the orthonormal basis of W and $W^\perp$, respectively. Let X' be the projection of X on the space W, that is $X' = S^T X$, and*

$$u^* = argmin \left\{ \mathcal{V}^2(u^T X', Y) : \|u\|_2 = 1 \right\}.$$

Then, Su^* is orthogonal to all the directions in $W^\perp$, and if we define a new subspace $K^\perp$ spanned by an orthonormal basis $\{Su^*\} \cup S^\perp$, then $W^\perp \subset K^\perp \subseteq W_X^\perp$.

A proof can be found in the appendix. As mentioned earlier, throughout this dissertation, we always relegate the proofs to the appendix whenever possible.

1.2.4 Independence Test

After each iteration of minimizing $\mathcal{V}^2(u^T X, Y)$, one needs to decide whether $u^T X$ and Y is independent or not, and whether we can proceed to the next step of looking for another

direction. The decision is made through the independence test of $u^T X$ and Y. The following two theorems, which are quoted from [20], form the theoretical foundation of the independence testing. Theorem 1.2.9 shows that when the sample size is large enough, the distribution of empirical distance covariance can be described as Gaussian distributions; Theorem 1.2.10 is about the range of the type I error of the independence test.

Theorem 1.2.9. *(Corollary 2 in [20]) If* $\mathbb{E}\left(|X|_p + |Y|_q\right) < \infty$*, then we have*

1. *If* X *and* Y *are independent,* $N\mathcal{V}_N^2/S_2(\mathbf{X},\mathbf{Y}) \xrightarrow[n\to\infty]{D} Q$ *where* $\mathbb{E}Q = 1$ *and* Q *is a nonnegative quadratic form of centered Gaussian random variables, defined as* $Q \overset{D}{=} \sum_{j=1}^{\infty} \lambda_j Z_j^2$ *where* Z_j*'s are independent standard normal random variables, and* λ_j*'s are nonnegative constants depending on the distribution of* (X, Y)*. Recall* S_2 *is defined in the Equation* (2.3)*.*

2. *If X and Y are dependent, then* $N\mathcal{V}_N^2/S_2(\mathbf{X},\mathbf{Y}) \xrightarrow[n\to\infty]{P} \infty$*.*

Theorem 1.2.10. *(Theorem 6 in [20]) Suppose the test rejects independence if*

$$N\mathcal{V}_N^2(\mathbf{X},\mathbf{Y}) > S_2(\mathbf{X},\mathbf{Y})\left(\Phi^{-1}(1-\frac{\alpha}{2})\right)^2, \tag{2.5}$$

where $\Phi(\cdot)$ *denotes the standard normal cumulative distribution function, and let* $\alpha(X, Y, n)$ *denote the achieved significance level of the test. If* $\mathbb{E}\left(|X|_p + |Y|_q\right) < \infty$*, then for all* $0 < \alpha \leq 0.215$*, we have*

1. $\lim_{n\to\infty} \alpha(X, Y, n) \leq \alpha$*, and*

2. $\sup_{X,Y}\left\{\lim_{n\to\infty} \alpha(X, Y, n) : \mathcal{V}(X, Y) = 0\right\} = \alpha.$

1.2.5 DISCA algorithm

Section 1.2.3 gives us an overview of how to find W_X and W_Y in the population point of view. Suppose we have $\mathbf{X} \in \mathbb{R}^{N\times p}, \mathbf{Y} \in \mathbb{R}^{N\times q}$, which are the samples of X and Y,

respectively. Each row represents one observation. Our strategy of estimating W_X and W_Y when $\mathbf{X}$ and $\mathbf{Y}$ are given can be summarized as follows.

Initialization: Let $X = \mathbf{X} \in \mathbb{R}^{N\times p}$, $Y = \mathbf{Y} \in \mathbb{R}^{N\times q}$, S_X be the set of orthonormal basis of space W_X with dimension d_X, S_Y be the set of orthonormal basis of space W_Y with dimension d_Y, $S_X^{\perp}$ be the set of orthonormal basis of $W_X^{\perp}$, and $S_Y^{\perp}$ be the set of orthonormal basis of $W_Y^{\perp}$. Initialize $S_X^{\perp} = S_Y^{\perp} = \emptyset$. Then correspondingly we have $S_X = I_p$, $S_Y = I_q$, $d_X = p$, $d_Y = q$.

Estimating W_X: Repeat the following steps until the condition in Step 3 is satisfied.

Step 1: Let X be the projection of X onto the subspace W_X, that is, $X \leftarrow XS_X$.

Step 2: Find $u \in S^{d_X-1}$ such that $\mathcal{V}_N^2(Xu, Y)$ is minimized. Suppose the solution is u^*.

Step 3: Calculate the squared empirical distance covariance, $\mathcal{V}_N^2(Xu^*, Y)$. If the condition

$$N \cdot \mathcal{V}_N^2(Xu^*, Y) > S_2(Xu^*, Y)\left(\Phi^{-1}(1-\frac{\alpha}{2})\right)^2, \text{ where } S_2 \text{ is defined in (2.3)}$$

is satisfied, stop here and the subspace spanned by S_X is the orthonormal basis of the final W_X. Otherwise, transform u^* into the original space $\mathbb{R}^p$, that is, $S_X u^*$, and then add $S_X u^*$ into the set $S_X^{\perp}$, and also update the dimension of the subspace W_X: $d_X \leftarrow d_X - 1$, and repeat the process from Step 1.

Estimating W_Y: Using the same strategy, one can compute for S_Y, the orthonormal basis of W_Y by switching X and Y in the previous procedure of finding W_X, with one change of using the $P_{W_X}X$ instead of X. In other words, we switch X and Y, and replace X by $P_{W_X}X$ in the procedure of estimating W_X until the stop condition is satisfied.

1.2.6 Estimating u^*

From the previous subsection, we can see that the key of our method is to find the solution, denoted as u^*, of

$$\min\{\mathcal{V}_N^2(\mathbf{X}u, \mathbf{Y}) : u \in S^{p-1}\}, \tag{2.6}$$

which can be formulated as Equation (2.7) as shown in Lemma 1.2.11.

Lemma 1.2.11. *Solving problem* (2.6) *is equivalent to solving*

$$\begin{aligned} &\underset{u\in\mathbb{R}^p}{\textit{min}} && \|M_+u\|_1 - \|M_-u\|_1 \\ &\textit{subject to:} && \|u\|_2 = 1, \end{aligned} \tag{2.7}$$

where

$$M_+ = \left[g_{ij}(X_i - X_j)^T\right]_{(i,j):g_{ij}>0, j>i} \in \mathbb{R}^{n_+\times p}, \textit{ where } n_+ = \left|\{(i,j) : g_{ij} > 0, j > i\}\right|,$$

$$M_- = \left[(-g_{ij})(X_i - X_j)^T\right]_{(i,j):g_{ij}<0, j>i} \in \mathbb{R}^{n_-\times p}, \textit{ where } n_- = \left|\{(i,j) : g_{ij} < 0, j > i\}\right|,$$

and g_{ij}*'s*$(i, j = 1, \cdots, n)$ *are defined as*

$$g_{ij} = |Y_i - Y_j|_q - \frac{1}{N}\sum_{k=1}^{N}|Y_i - Y_k|_q - \frac{1}{N}\sum_{k=1}^{N}|Y_j - Y_k|_q + \frac{1}{N^2}\sum_{k,l=1}^{N}|Y_k - Y_l|_q.$$

Note that the problem (2.7) is a non-convex problem with a quadratic constraint. We first adopt the penalty method to transform the above problem into an unconstraint problem, and then apply the difference-of-convex algorithm (DCA) to find a local solution. Details can be found in Section 1.4.

1.3 Theoretical Results

In this section, we establish the consistency properties of our procedure in Section 1.3.1. In Section 1.3.2, we articulate the advantages of our method in comparison with CCA-, SDR-, and DCS- types of competitors.

1.3.1 Consistency Properties

Before showing that the procedure in Section 1.2.5 will converge to the true W_X, we verify that the solution of (2.7) is convergent to a unit vector in W_X in each iteration of our method.

Lemma 1.3.1. *In general, under the Assumption 1.2.5 and 1.2.6, assume subspace $W' \subseteq \mathbb{R}^p$ satisfying $(W')^\perp \subset W_X^\perp$, and U is an orthonormal basis of W'. $X' = U^T X$. (Note that W' could be $\mathbb{R}^p$, which leads to $(W')^\perp = \emptyset$. It still satisfies the condition $(W')^\perp \subset W_X^\perp$. In this case U can be chosen as the identity matrix, and $X' = X$.) Let u be the vector with positive first nonzero element, which is also a solution of the following problem:*

$$u = \textit{argmin} \left\{ \|M_+ u\|_1 - \|M_- u\|_1 : \|u\|_2 = 1 \right\}.$$

Then, there exists some u^* such that we have $u \to u^*$ as $N \to \infty$, where u^* is a vector with positive first nonzero element satisfying

$$u^* = \textit{argmin} \left\{ \mathcal{V}^2(u^T X', Y) : \|u\|_2 = 1 \right\}.$$

Next we will show that the subspace obtained by our method will converge to the real subspace. Before that, we need the definition of the distance between two subspace. Suppose W_1 and W_2 are two equal-dimensional subspaces in $\mathbb{R}^n$. The distance between them can be defined as in [25]:

$$\text{dist}(W_1, W_2) = \|P_1 - P_2\|_2,$$

where P_i is the orthogonal projection onto W_i (for $i = 1, 2$). The following theorem is important in the calculation of the subspace distances.

Theorem 1.3.2. *(Theorem 2.5.1 in [25]) Suppose that $A = [A_1, A_2], B = [B_1, B_2]$ are n-by-n orthogonal matrices. If we have $W_1 = span(A_1)$, and $W_2 = span(B_1)$, then we have*

$$dist(W_1, W_2) = \|A_1^T B_2\|_2 = \|A_2^T B_1\|_2.$$

Given the above theorem, we can show that DISCA can identify the true underlying dependency structure when the sample size goes to infinity:

Theorem 1.3.3. *Suppose the subspace estimate of W_X by DISCA is $\widehat{W}_X$, and $dim(W_X) = dim(\widehat{W}_X)$. Then we have*

$$dist(W_X, \widehat{W}_X) \to 0, \text{ as } N \to \infty.$$

Notice that Theorem 1.3.3 assumes that the dimension of the space $\widehat{W}_X$ is equal to the dimension of the true space W_X, which requires that the iteration stops at a right time. The probability that we will get the right dimension of W_X is guaranteed by the following theorem:

Theorem 1.3.4. *Suppose at iteration t, the dimension of the estimate of the subspace $\widehat{W}_X^{(t)}$ is equal to the dimension of the true subspace W_X. Let $P_N^{(t)}$ be the probability that the procedure will stop at the right iteration t. For all $0.785 < \gamma < 1$, we have*

1. $\lim_{N\to\infty} P_N^{(t)} \geq \gamma$;
2. $\inf_{u^T X \perp\!\!\!\perp Y} \left\{ \lim_{N\to\infty} P_N^{(t)} \right\} = \gamma$.

1.3.2 Comparison with Existing Methods

DISCA can reduce dimensionality in some circumstances where CCA , SDR, and DCS cannot do. Here we give a detailed explanation of our comparison with CCA, SDR, and

DCS methods.

As stated in Section 2.1, the canonical variables appear in pairs in CCA methods. Although the dimension of X and Y could be different, we will still reduce the dimensionality for both X and Y to two equal-dimensional spaces. No matter how advanced CCA has been developed nowadays, the optimization form has not changed. Therefore, the limitation still exists.

As for SDR, restricted by the regression setting, X and Y are not equally treated, which makes it not suitable to the problem setting in this chapter. If we omit the operation complexity, a disputation would be to take Y as responses to do SDR with X to get the dimensional reduction subspace for X and then switch the position of X and Y to do the same thing as above to get the dimensional reduction subspace of Y. But this still cannot work when at least two of the Y_j's is relevant. Suppose two random vectors $X = (X_1, \ldots, X_p)^T \in \mathbb{R}^p, Y = (Y_1, \ldots, Y_q)^T \in \mathbb{R}^q$ satisfy

$$Y_j = \sum_{i=1}^{p} X_i + \epsilon_j, j = 1, \ldots, q,$$

where ϵ_j's are random noises independent of X. Then the dimensional reduction subspace for X, that is, W_X, is the subspace spanned by $(\frac{1}{\sqrt{p}}, \frac{1}{\sqrt{p}}, \ldots, \frac{1}{\sqrt{p}})^T$, and that for Y, that is, W_Y, is the subspace spanned by $(\frac{1}{\sqrt{q}}, \frac{1}{\sqrt{q}}, \ldots, \frac{1}{\sqrt{q}})^T$. The dimension of both of the two subspaces is 1. But if we apply SDR for X with respect to Y, and Y with respect to X, we can get the true dimension reduction subspace W_X, but fail to get the true W_Y because of the non-exchangeability in the regression setting.

The limit of DCS is not too critical compared with the other two. In the stage of determining how many dimensions should be kept by bootstrap, it requires the randomness of the subspaces, which causes DCS cannot handle when at least one of the two random vectors cannot be dimensionally reduced.

Above all, we construct one simple counterexample that all the three do not work

but DISCA still can. Suppose two random vectors $X = (X_1, \ldots, X_p)^T \in \mathbb{R}^p, Y = (Y_1, \ldots, Y_q)^T \in \mathbb{R}^q$ satisfy

$$Y_j = f_j\left(\sum_{i=1}^{p} X_i\right) + \epsilon_j, j = 1, \ldots, q,$$

where f_j's are q different types of functions, and ϵ_j's are random noises independent of X. The true dimensional reduction subspace for X would be the one spanned by $(\frac{1}{\sqrt{p}}, \frac{1}{\sqrt{p}}, \ldots, \frac{1}{\sqrt{p}})^T$, and there is nothing we can do to reduce dimensionality for Y. In other words, $\dim(W_X) = 1, \dim(W_Y) = q$. CCA-related methods are incapable of detecting this kind of structure because if CCA stops after one iteration, it will give us only a pair of directions (u, v) in which u might contain all the information we would like to know in X but v only has one dimension of the whole p dimension space; if CCA stops after q iterations (assuming $q < p$), there would be too much redundant information for X. Because of the regression setting, SDR is unable to work well. Since Y cannot be dimensionally reduced, DCS is ineffective. The simulation results regarding the counterexample for the comparison with CCA, SDR, and DCS are in Section 1.5.1.

1.4 Applying DCA for estimating u^*

As mentioned in Section 1.2.6, the problem (2.7), which we eventually need to solve, is a non-convex problem. Considering its special form (Lemma 1.2.11), we apply DCA to do the calculation.

A review of the difference-of-convex algorithms is provided in Section 1.4.1. The corresponding minimization problem is presented in Section 1.4.2. The adoption of the ADMM to solve a subproblem is furnished in Section 1.4.3, followed by the convergence analysis in Section 3.5.

1.4.1 Review of DCA

Difference-of-Convex Algorithm (DCA) [24] is used to solve the optimization problems that are related to DC (difference of convex) functions, which is defined below.

Definition 1.4.1. *(DC function) Let f be a real-valued function mapping $\mathbb{R}^n$ to $\mathbb{R}$. Then f is a DC function if there exist convex functions, $g, h : \mathbb{R}^n \to \mathbb{R}$, such that f can be decomposed as the difference between g and h:*

$$f(x) = g(x) - h(x), \forall x \in \mathbb{R}^n.$$

Difference of Convex Algorithm (DCA) is aimed to solve the following problem:

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & f_0(x) \\ \text{subject to:} \quad & f_i(x) \leq 0, i = 1, \cdots, m, \end{aligned} \tag{4.8}$$

where $f_i : \mathbb{R}^n \to \mathbb{R}$ is a differentiable DC function for $i = 0, ..., m$.

Let $\partial f(x)$ be the subgradient of f at x, and $f^*(y)$ be the conjugate of $f(x)$. We have the following lemma.

Lemma 1.4.2. *If $f : \mathbb{R}^p \to \mathbb{R}$ is lower semi-continuous and convex, then*

$$x \in \partial f^*(y) \Longleftrightarrow x \in \textit{argmax} \left\{ y^T x - f(x) : x \in \mathbb{R}^p \right\}.$$

Algorithm 1 shows the details of the DCA algorithm.

1.4.2 Minimizing $\mathcal{V}_N^2(\mathbf{X}u, \mathbf{Y})$

As mentioned before, one can apply the augmented Lagrangian method to transform the original problem (2.7) into an unconstrained problem. Assume we have $\xi^{(t)} \geq 0$. Here we use t as the count of iterations. The problem (2.7) can be solved as a series of unconstrained

Algorithm 1 Difference of Convex Algorithm (DCA) [24]

Initialization: choose u_0, α, β;

1: **for** $k \in \mathbb{N}$ **do**
2: Choose $y_k \in \partial h(u_k)$;
3: Choose $u_{k+1} \in \partial g^*(y_k)$;
4: **if** $\max_i \left\{ \left| \frac{(u_{k+1}-u_k)_i}{(u_k)_i} \right| \right\} < e$ **then**
5: **return** u_{k+1}.
6: **end if**
7: **end for**

minimization problem which is to minimize the following function

$$\begin{aligned} L(u;\xi^{(t)},\psi^{(t)}) &= \|M_+u\|_1 - \|M_-u\|_1 + \psi^{(t)}\left(\|u\|_2 - 1\right) + \frac{\xi^{(t)}}{2}(\|u\|_2 - 1)^2 \\ &= \left(\frac{\xi^{(t)}}{2}u^Tu + \|M_+u\|_1\right) - \left(\|M_-u\|_1 + \left(\xi^{(t)} - \psi^{(t)}\right)\|u\|_2\right) + \frac{\xi^{(t)}}{2} - \psi^{(t)}. \end{aligned}$$

Let $g(u;\xi^{(t)}) = \frac{\xi^{(t)}}{2}u^Tu + \|M_+u\|_1, h(u;\xi^{(t)},\psi^{(t)}) = \|M_-u\|_1 + \left(\xi^{(t)} - \psi^{(t)}\right)\|u\|_2$. Then we have

$$L(u;\xi^{(t)},\psi^{(t)}) = g(u;\xi^{(t)}) - h(u;\xi^{(t)},\psi^{(t)}) + \frac{\xi^{(t)}}{2} - \psi^{(t)}. \tag{4.9}$$

In each iteration t, one minimizes the augmented Lagrangian function, and then update the ψ to be $\psi^{(t)} + \xi^{(t)}(\|u\|_2 - 1)$, and increase $\xi^{(t)}$ gradually. Per the updating rule of augmented Lagrangian method, eventually ξ will go beyond some threshold, and ψ will converge to the true Lagrangian multiplier. So if we choose ξ to be large enough, then $\xi - \psi > 0$ is satisfied. Therefore, both $g(u;\xi^{(t)})$ and $h(u;\xi^{(t)},\psi^{(t)})$ are convex, and we can now apply the Difference-of-Convex Algorithm (DCA) on it by omitting the constant term $\frac{\xi^{(t)}}{2} - \psi^{(t)}$.

From Algorithm 1, we need to know $\partial h(u_k;\xi^{(t)},\psi^{(t)})$ and $\partial g^*(y_k;\xi^{(t)})$. By calculation, we can get

$$\partial h(u_k;\xi^{(t)},\psi^{(t)}) = \begin{cases} M_-^T\partial\|M_-u_k\|_1 + \left(\xi^{(t)} - \psi^{(t)}\right)\frac{u_k}{\|u_k\|_2}, & \text{if } u_k \neq 0; \\ M_-^T\partial\|M_-u_k\|_1 + \left(\xi^{(t)} - \psi^{(t)}\right)\{w : \|w\|_2 \leq 1\}, & \text{if } u_k = 0, \end{cases}$$

where each entry of $\partial\|\cdot\|_1$ is defined as

$$(\partial\|x\|_1)_i = \begin{cases} 1, & \text{if } x_i > 0; \\ (-1,1), & \text{if } x_i = 0; \\ -1, & \text{if } x_i < 0. \end{cases}$$

Applying Lemma 1.4.2 on $g(u;\xi^{(t)})$, we can get

$$\begin{aligned} \partial g^*(y_k;\xi^{(t)}) \in & \ \text{argmax}\left\{y_k^T u - g(u) : u \in \mathbb{R}^p\right\} \\ =& \text{argmin}\left\{g(u) - y_k^T u : u \in \mathbb{R}^p\right\} \\ =& \text{argmin}\left\{\frac{\xi^{(t)}}{2}u^T u + \|M_+ u\|_1 - y_k^T u : u \in \mathbb{R}^p\right\}. \end{aligned}$$

Overall, our algorithm for getting the solution of minimizing function (4.9) can be summarized as in Algorithm 2.

Algorithm 2 DCA for minimizing (4.9) in iteration t

Initialization: choose $u^{(0)}$;

1: **for** $k \in \mathbb{N}$ **do**

2: Let $u_0 = u^{(t)}$, and

$$y_k = \begin{cases} M_-^T \partial\|M_- u_k\|_1 + \left(\xi^{(t)} - \psi^{(t)}\right)\frac{u_k}{\|u_k\|_2}, & \text{if } u_k \neq 0; \\ M_-^T \partial\|M_- u_k\|_1 + \left(\xi^{(t)} - \psi^{(t)}\right)\{w : \|w\|_2 \leq 1\}, & \text{if } u_k = 0. \end{cases}$$

3: $u_{k+1} = \text{argmin}\left\{\frac{\xi^{(t)}}{2}u^T u + \|M_+ u\|_1 - y_k^T u : u \in \mathbb{R}^p\right\}$.

4: **if** $\max_i \left\{\left|\frac{(u_{k+1}-u_k)_i}{(u_k)_i}\right|\right\} < e$ **then**

5: **return** u_{k+1} as $u^{(t+1)}$.

6: **end if**

7: **end for**

1.4.3 Solving the Subproblem

The line 4 of Algorithm 2 needs the solution of the following:

$$\min_{u\in\mathbb{R}^p}\left\{\frac{\xi^{(t)}}{2}u^Tu+\|M_+u\|_1-y_k^Tu\right\}. \tag{4.10}$$

It is a convex programming problem, and can be solved by a lot of methods, such as the interior-point method. As Alternating Direction Method of Multipliers (ADMM) [21] is efficient in calculation, we use ADMM rather than others. ADMM solves the following problem:

$$\begin{aligned}\min_{x\in\mathbb{R}^n,z\in\mathbb{R}^m}\quad & f(x)+g(z)\\ \text{subject to:}\quad & Ax+Bz=c.\end{aligned}$$

The trick is to split the variable in the problem into two seperate parts. In our case, recall the number of rows in M_+ is n_+, and then (4.10) can be rewritten as

$$\begin{aligned}\min_{u\in\mathbb{R}^p,z\in\mathbb{R}^{n_+}}\quad & \frac{\xi^{(t)}}{2}u^Tu+\|z\|_1-y_k^Tu\\ \text{subject to:}\quad & M_+u-z=0.\end{aligned} \tag{4.11}$$

The augmented Lagrangian of (4.11) is

$$L_\rho(u,z,v)=\frac{\xi^{(t)}}{2}u^Tu+\|z\|_1-y_k^Tu+v^T(M_+u-z)+\frac{\rho}{2}\|M_+u-z\|_2^2,$$

where v is the Lagrangian multiplier, and $\rho>0$ is the penalty parameter.

According to [21], we need to update u, z, and v as follows:

$$\begin{cases}u_{l+1}=\text{argmin}L_\rho(u,z_l,v_l);\\ z_{l+1}=\text{argmin}L_\rho(u_{l+1},z,v_l);\\ v_{l+1}=v_l+\rho(M_+u_{l+1}-z_{l+1}).\end{cases} \tag{4.12}$$

Through calculations, the results in our case are included in the following lemma.

Lemma 1.4.3. *The update rules of u and z for solving problem* (4.11) *are*

$$\begin{aligned} u_{l+1} &= \left(\xi^{(t)} I_p + \rho M_+^T M_+\right)^{-1} \left(y_k + M_+^T(\rho z_l - v_l)\right); \\ z_{l+1} &= S(\frac{1}{\rho} v_l + M_+ u_{l+1}, \frac{1}{\rho}), \end{aligned}$$

where the soft thresholding operator is defined as $S(x, y) \in \mathbb{R}^p$,

$$(S(x, y))_i = \textit{sgn}(x_i) \max\{|x_i| - y, 0\}.$$

If we define $r_l = M_+ u_l - z_l$, $s_l = \rho M_+^T(z_l - z_{l-1})$, based on [21], we have the following stop criterion:

$$\|r_l\|_2 \leq \sqrt{n_+}\epsilon^{abs} + \epsilon^{rel} \max\left\{\|M_+ u_l\|_2, \|z_l\|_2\right\},$$

and

$$\|s_l\|_2 \leq \sqrt{p}\epsilon^{abs} + \epsilon^{rel}\|M_+^T v_l\|_2,$$

where ϵ^{abs} is an absolute tolerance, and ϵ^{rel} is a relative tolerance.

To sum up, our algorithm can be summarized as in Algorithm 3.

Algorithm 3 ADMM for updating u_{k+1} in the loop of DCA

Initialization: choose z_0, v_0;

1: **for** $l \in \mathbb{N}$ **do**
2: $u_{l+1} = \left(\xi^{(t)} I_p + \rho M_+^T M_+\right)^{-1} \left(y_k + M_+^T(\rho z_l - v_l)\right)$;
3: $z_{l+1} = S(\frac{1}{\rho} v_l + M_+ u_{l+1}, \frac{1}{\rho})$;
4: $v_{l+1} = v_l + \rho(M_+ u_{l+1} - z_{l+1})$;
5: **if** $\|r_l\|_2 \leq \sqrt{n_+}\epsilon^{abs} + \epsilon^{rel} \max\left\{\|M_+ u_l\|_2, \|z_l\|_2\right\}$, and $\|s_l\|_2 \leq \sqrt{p}\epsilon^{abs} + \epsilon^{rel}\|M_+^T v_l\|_2$, **then**
6: **return** u_{l+1}.
7: **end if**
8: **end for**

1.4.4 Convergence Analysis

It is not guaranteed that all difference of convex problems are convergent. So convergence analysis is provided for our algorithm. We need the following lemma to proceed to our main theorem stating that our algorithm will give us a stationary solution in each iteration t.

Lemma 1.4.4. *Let $\{u_k\}$ be the sequence generated by our algorithm 2 in a specific iteration t. For all $k \in \mathbb{N}$, we have*

$$L(u_k; \xi^{(t)}) - L(u_{k+1}; \xi^{(t)}) \geq \frac{\xi^{(t)}}{2} \|u_{k+1} - u_k\|_2^2 \geq 0.$$

Theorem 1.4.5. *Let $\{u_k\}$ be the sequence generated by our algorithm 2 in a specific iteration t. The followings are true.*

1. *$\{u_k\}$ is bounded, and $\|u_{k+1} - u_k\|_2 \to 0$ as $k \to +\infty$.*
2. *Any nonzero limit point $u^{(t)}$ of $\{u_k\}$ satisfies the first-order optimality condition*

$$0 \in M_+^T \partial \|M_+ u^{(t)}\|_1 - M_-^T \partial \|M_- u^{(t)}\|_1 + \xi^{(t)} u^{(t)} - \left(\xi^{(t)} - \psi^{(t)}\right) \frac{u^{(t)}}{\|u^{(t)}\|_2}.$$

   This indicates that $u^{(t)}$ is a stationary point.

3. *For certain k satisfying $\|u_k - u^{(t)}\| < \frac{1}{k}$, then we have $\left|L(u_k; \xi^{(t)}) - L(u^{(t)}; \xi^{(t)})\right| = O(\frac{1}{k})$.*

As before, all proofs are relegated to the appendix.

1.5 Simulation Studies

Denote the subspaces generated by DISCA or other methods as $\widehat{W}_X$, and $\widehat{W}_Y$, respectively, the true subspaces are denoted as W_X and W_Y. In Section 1.5.1, we give the simulation

results for the counterexample to show that only DISCA can work; and then in Section 1.5.2 we provide examples showing that DISCA can handle both the discrete and the heavy-tailed cases. A real data example is provided in Section 1.5.3. We choose both ϵ_{abs} and ϵ_{rel} to be 10^{-3} in this section.

1.5.1 Counterexample Simulation

CCA can only get pairs of canonical variables which results in the disability of performing the correct dimension reduction when the dimensions of the reduced subspaces are not equal; SDR may not perform well because of the non-exchangeability of responses and predictors; DCS is not working when it is not necessary to reduce dimensions of at least one of the random vectors. Section 1.3 already constructed a general case of a simple counterexample for comparison with the above, and according to that example, the following is a simple but clear example structure that can demonstrate this point:

Suppose $X \in \mathbb{R}^3$ and $Y \in \mathbb{R}^2$. We have $X = (X_1, X_2, X_3)^T, Y = (Y_1, Y_2)^T$. In addition, we have

$$\begin{aligned} X &\sim N(\mathbf{0}, I_3), \\ Y_1 &= \sum_{i=1}^{3} X_i + 0.01\epsilon_1, \\ Y_2 &= \left(\sum_{i=1}^{3} X_i\right)^2 + 0.01\epsilon_2, \end{aligned}$$

where ϵ_1, ϵ_2 are i.i.d following the standard normal distribution.

Recall the definition of W_X and W_Y: their orthogonal complement projections, $P_{W_X^\perp}$ and $P_{W_Y^\perp}$ satisfy $P_{W_X^\perp} X \perp\!\!\!\perp Y$ and $P_{W_Y^\perp} Y \perp\!\!\!\perp X$. Then the anticipated reduced subspace of X, that is W_X, is supposed to be the subspace spanned by $(\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}})^T$; as there are other random factors ϵ_1 and ϵ_2 in addition to the X_i's, there is no way to reduce the dimension of Y, so the anticipated reduced subspace of Y, that is W_Y, is supposed to be the subspace spanned by $(1, 0)^T$ and $(0, 1)^T$.

We simulate samples with sizes $N = 50, 100, 150, 200$ for 500 times each. As the subspace W_Y was accurately found each time without error, we will focus on $\widehat{W}_X$. Table 1.1 shows how many times DISCA reduced X into 0,1,2 dimension subspaces respectively for different N; Figure 1.1 is the box plot for the distances between the $\widehat{W}_X$ produced by DISCA and the true subspace W_X. From the table and figure we can tell that the performance of DISCA (both the accuracy of the dimension of the reduced subspace and the subspace itself) is improved as the sample size increases.

$\dim(\widehat{W}_X)$	0	1	2
N=50	6	492	2
N=100	1	496	3
N=150	0	499	1
N=200	0	499	1

Table 1.1: Table of the dimension of $\widehat{W}_X$

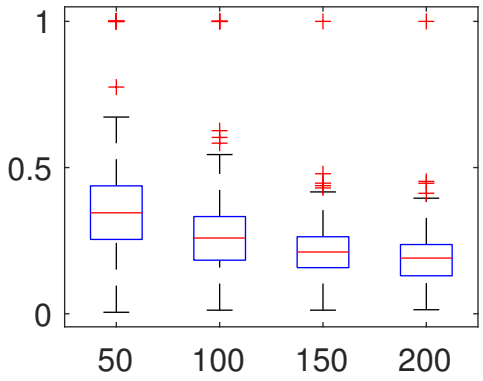


Figure 1.1: Boxplot of $dist(\widehat{W}_X, W_X)$

1.5.2 Comparison with Existing Methods

The following are examples of the performance of DISCA and other existing methods – CCA and DCS. As DCS needs to perform bootstrap to determine the dimension of the reduced subspaces, which is extremely time consuming, here we just assume the bootstrap gives the correct dimension and used the correct number to find the subspaces. Similarly, we need to know how many pairs of canonical variables are significant so here we select the correct number of pairs as well. Notice that when performing DISCA, we did not give any prior knowledge of the subspace dimensions.

We constructed three different types of examples for illustration. Example 1.5.1 is a continuous distribution case, which seems similar to the one in the above subsection, but it actually not: the covariance matrix is more complicated; the dimension of Y is changed;

the relation between X and Y is changed as well (including independent relation as well as the linear and polynomial nonlinear relations). With these changes, DCS now works while CCA is still not applicable. Example 1.5.2 is a discrete distribution case with independent relation and polynomial nonlinear relation between X and Y. Example 1.5.3 is a heavy-tailed distribution case with complicated nonlinear relation between X and Y.

In each example, we simulate $N = 50, 100, 150, 200$ for 1000 times. Similar to the above section, we calculate the distances between the subspaces obtained from different dimension reduction methods, $\widehat{W_X}$ or $\widehat{W_Y}$, and the true subspaces W_X and W_Y, and draw boxplots for each scenarios.

Example 1.5.1. *(Normal distribution example)*

Suppose $X \in \mathbb{R}^3$ *and* $Y \in \mathbb{R}^3$.

$$X = (X_1, X_2, X_3)^T, Y = (Y_1, Y_2, Y_3)^T.$$

X follows the multivariate normal distribution with zero mean and covariance matrix

$$\Sigma = \begin{pmatrix} 1 & 0.5 & 0.5 \\ 0.5 & 1 & 0.5 \\ 0.5 & 0.5 & 1 \end{pmatrix}.$$

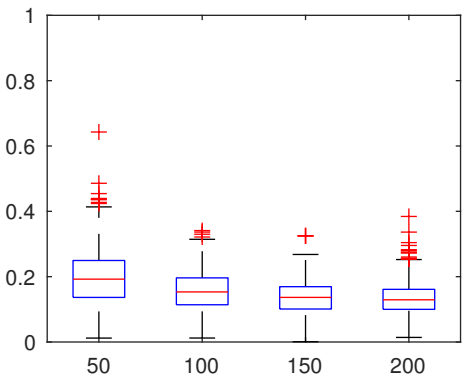
Y_1 and Y_2 are independent, satisfying $Y_1 = X_1 + X_2 + X_3 + 0.01\epsilon_1, Y_2 = (X_1 + X_2 + X_3)^2 + 0.01\epsilon_2, Y_3 \sim N(0, 1)$, *where ϵ_1, ϵ_2 are i.i.d. from the standard normal distribution.*

The anticipated results would be

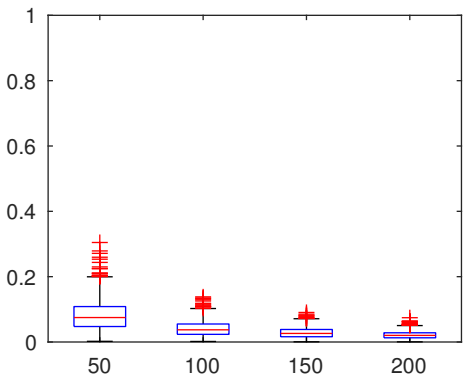
$$W_X = \textit{span}\left\{(\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}})^T\right\}; \quad W_Y = \textit{span}\left\{(1, 0, 0)^T, (0, 1, 0)^T\right\}.$$

The calculation results are as in Figure 1.2.

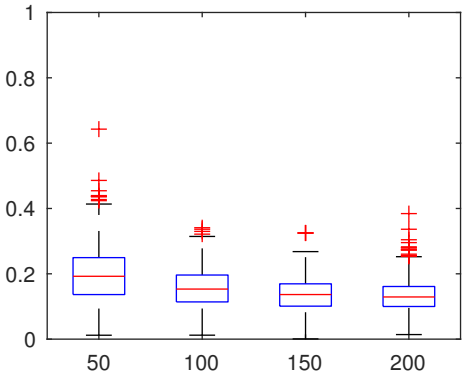
Example 1.5.2. *(Discrete distribution example)*



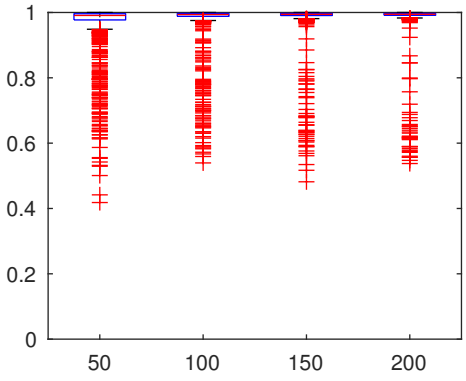
Boxplot of $dist(\widehat{W}_X, W_X)$ by *DISCA*



Boxplot of $dist(\widehat{W}_X, W_X)$ by *DCS*



Boxplot of $dist(\widehat{W}_Y, W_Y)$ by *DISCA*



Boxplot of $dist(\widehat{W}_Y, W_Y)$ by *DCS*

Figure 1.2: The above figures are the results of Example 1: figures on the first row are the boxplots of $dist(\widehat{W}_X, W_X)$ obtained by DISCA and DCS respectively; figures on the bottom row are the boxplots of $dist(\widehat{W}_Y, W_Y)$ obtained by DISCA and DCS respectively. The x-axis represents $N = 50, 100, 150, 200$.

Suppose $X \in \mathbb{R}^3$ *and* $Y \in \mathbb{R}^2$.

$$X = (X_1, X_2, X_3)^T, Y = (Y_1, Y_2)^T.$$

$$X_i \sim B(10, 0.5), i.i.d, i = 1, 2, 3; Y_1 = (X_1 + X_2 + X_3)^2 + 0.01\epsilon, Y_2 \sim B(10, 0.35)$$

where ϵ *is from the standard normal distribution.*

The anticipated results would be

$$W_X = \textit{span}\left\{(\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}})^T\right\}; \quad W_Y = \textit{span}\left\{(1, 0)^T\right\}.$$

The calculation results are as in Figure 1.3.

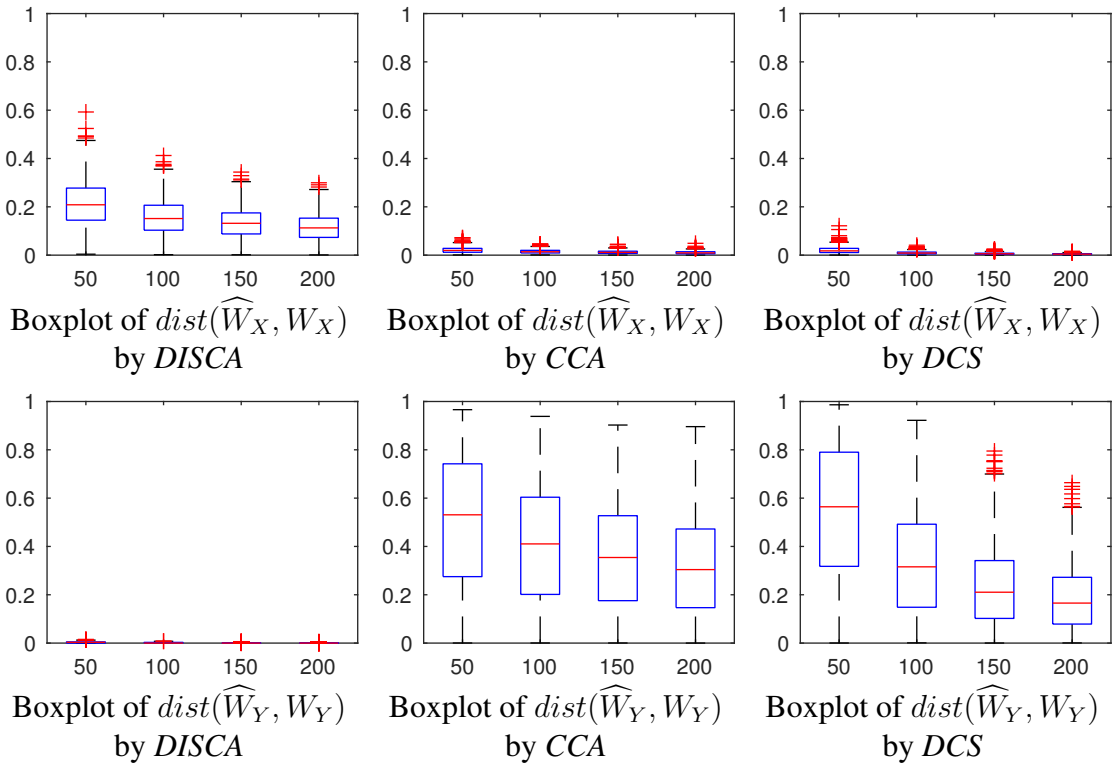


Figure 1.3: The above figures are the results of Example 2: the figures in the first row are the boxplots of $dist(\widehat{W}_X, W_X)$ obtained by DISCA, CCA, and DCS respectively; the figures in the bottom row are the boxplots of $dist(\widehat{W}_Y, W_Y)$ obtained by DISCA, CCA, and DCS respectively. The x-axis represents $N = 50, 100, 150, 200$.

Example 1.5.3. *(Heavy-tailed distribution example)*

Suppose $X \in \mathbb{R}^3$ *and* $Y \in \mathbb{R}^2$.

$$X = (X_1, X_2, X_3)^T, Y = (Y_1, Y_2)^T.$$

$X_i \sim t(2), i.i.d, i = 1, 2, 3; Y_j = tanh\,(X_1 + X_2 + X_3) + 0.01\epsilon_j, j = 1, 2$ *where* ϵ_1, ϵ_2 *are from the standard normal distribution.*

The anticipated results would be

$$W_X = span\left\{(\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}})^T\right\}; \quad W_Y = span\left\{(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})^T\right\}.$$

The calculation results are as in Figure 1.4.

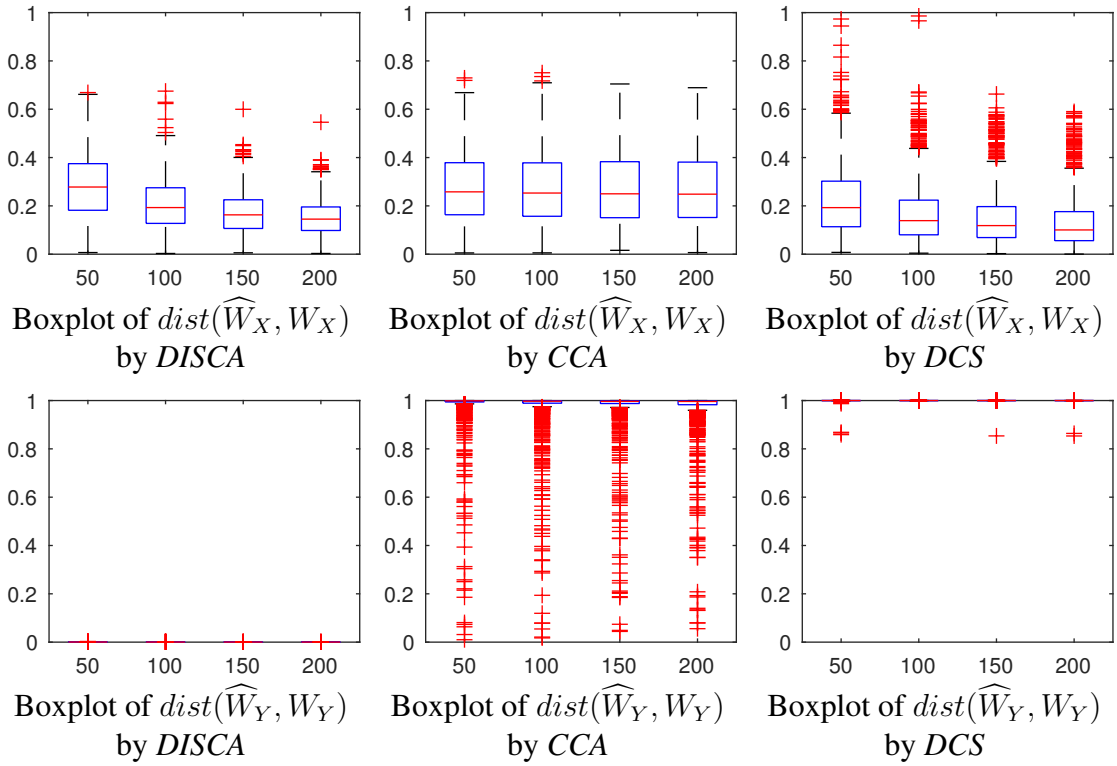


Figure 1.4: The above figures are the results of Example 3: the figures in the first row are the boxplots of $dist(\widehat{W}_X, W_X)$ obtained by DISCA, CCA, and DCS respectively; the figures in the second row are the boxplots of $dist(\widehat{W}_Y, W_Y)$ obtained by DISCA, CCA, and DCS respectively. The x-axis represents $N = 50, 100, 150, 200$.

From the above results we can see that DISCA performs well for both normal and heavy-tailed distribution, discrete and continuous distribution; the performance is improved as the sample size increases. It is not surprising that CCA and DCS performs slightly better than DISCA for the the dimension reduction of the linear relations for normal or nearly-normal distributions (That is, the dimension reduction for W_X in Example 1.5.1 and 1.5.2), since they are designed to detect the linear relations for normal or nearly normal distributions. For heavy-tailed distribution (Example 1.5.3) and nonlinear relations (That is, Example 1.5.3 as well as dimension reduction for W_Y in Example 1.5.1 and 1.5.2), however, DISCA shows strong advantages compared with the others. The simulation results confirm DISCA is more powerful in the scenarios involved with nonlinear relations and

heavy-tailed distributions.

1.5.3 LA Pollution-Mortality Study (1970-1979)

In this section, we use a real dataset to demonstrate our method. This data was first studied by [36], and was also studied by [19]. It contains 11 series of daily measurements in Los Angeles County from the year 1970 to 1979. The first three columns are three different kinds of mortality of all deaths of LA residents, LA nonresidents, and LA residents in other localities; The fourth and fifth columns are two weather measurements of maximum daily temperature and average relative humidity over five different monitoring stations; The next six columns are pollutants measurements of the average of their daily maxima at six monitoring stations. As in [36], we use the weekly data instead of the daily data to perform the analysis. The number of observations is 508.

Mortality (Y)	1. Total Mortality Y_1 (tmort) 2. Respiratory Mortality Y_2 (rmort) 3. Cardiovascular Mortality Y_3 (cmort)
Weather	4. Temperature X_1 (temp) 5. Relative Humidity X_2 (rh)
Pollutant	6. Carbon Monoxide X_3 (co) 7. Sulfur Dioxide X_4 (so2) 8. Nitrogen Dioxide X_5 (no2) 9. Hydrocarbons X_6 (hycarb) 10. Ozone X_7 (o3) 11. Particulates X_8 (part)

Table 1.2: Summary of the LA Pollution-Mortality Data

Let Y be the vector containing the three different kinds of mortality indices, and $X = (X_1, X_2, \ldots, X_8)^T$ be the weather and pollutant indices as illustrated in Table 1.2. The aim is to find the related parts of X and Y. We apply DISCA on the dataset and get the corresponding basis of W_X and W_Y subspaces, U and V. There is no dimension reduction for Y, and the details of U are summarized in Table 1.3. Notice that three measurements in X are dominant: hydrocarbons, ozone, and the particulates. In fact, if we do varimax

rotation for this matrix, we will get a matrix with the three bold positions being 1 and all the others being 0. From the results we have the following observations:

dimension	temp	rh	co	so2	no2	hycarb	o3	part
1	-0.1517	-0.1739	0.1216	-0.0111	0.1381	**0.9549**	0.0088	-0.0292
2	0.1729	0.0245	-0.2098	-0.2793	-0.1191	0.0662	**0.9082**	0.0641
3	-0.2504	-0.1255	0.1096	-0.0034	-0.0644	-0.0382	0.0025	**0.9507**

Table 1.3: DISCA reduced the 8-dimensional space of X into a 3-dimensional subspace, with basis vectors shown as the rows in the above table.

- Not only we can conclude the weather factors such as temperature are not relevant to mortality, but also we can say that the hydrocarbons, ozone, and particulates are three most influential pollutants related to mortality during the 10-year period.
- Another observation is that although the three different kinds of mortality seem similar, but as they cannot be reduced to a smaller subspace by projecting Y on some linear subspace, there may exist complicated relationships among the three and they cannot be simply represented.
- Compared with the results in [19], the results obtained by DISCA is more explainable as our results show explicitly which three components are important while their results are some complicated linear combination of the variables.

1.6 Conclusion

As we discussed above, dimension reduction is an important topic especially for multi-dimensional data. The existing dimension reduction methods cannot cover all the situations more or less. In this chapter, we propose a new dimension reduction method, DISCA, to address the issues caused by other methods, and it is strongly encouraged especially when the dependency structure involving complicated nonlinear relations and non-normal distributions. Besides, we have the computational advantage over the DCS method in [19], as their method need apply bootstrap to first determine the dimension of both W_X and W_Y,

which leads to $(p-1)(q-1)B$ times computation (B is the bootstrap times, which is usually large) while our method only performs once.

Furthermore, we presented the theoretical desirable properties of our method, and guaranteed the convergence of our algorithm in theory. Our simulation studies strongly support our method and theory results, from one dimension to multi-dimension reduction, normal to heavy-tailed distributions, and dimension reduction to no dimension reduction.

In future work, we would like to study the sparsity of DISCA since the significant directions obtained from DISCA often has one or two elements that are much larger than the others. Another potential direction is that the distance error seems to have a distribution pattern, and studying it might help us to further understand the performance of DISCA as well.

CHAPTER 2
OPTIMAL PROJECTIONS IN THE DISTANCE-BASED STATISTICAL METHODS

2.1 Introduction

Distances are very important in statistics: a class of hypotheses testing methods are based on distances, such as the energy statistics [38], the distance covariance [20, 46, 47], and many others. This type of testing statistics usually belong to the class of U-statistics or the V-statistics [50, 49, 48], which require the calculation of all pairwise distances within the sample. When variables are univariate, assuming the sample size is m, both [40] and [52] proposed fast algorithms with computational complexity $O(m\log(m))$ where m is the sample size. Recall that the computational complexity is $O(m^2)$ when the statistics are computed directly based on their definitions. When variables are multivariate, especially when they are high-dimensional, the calculation of the pairwise distances among these multivariate variables can not be implemented directly by the algorithm in [40], and therefore becomes a potential bottleneck. Our chapter is aimed at reducing the computation complexity in the multivariate case by projecting the variables along a set of pre-specified optimal directions. When the number of pre-specified optimal directions $n \ll m/\log(m)$, computational savings can be achieved, since the computational complexity is $O(nm \cdot \log(m))$, which would be less than $O(m^2)$.

We use the energy distances [38] as an example to solidify our motivation. The energy statistic is used to test the equality between two distributions. More precisely, suppose $X_1, ..., X_{n_1} \in \mathbb{R}^p, p \geq 1$ are independent and identically distributed (i.i.d.), sampled from the distribution F_X, and $Y_1, ..., Y_{n_2} \in \mathbb{R}^p$ are i.i.d., sampled from the distribution F_Y. The two-sample test statistic (also called the energy statistic) for testing the two-sample hypoth-

esis

$$H_0 : F_X = F_Y$$

is defined as [38]:

$$\mathcal{E}_{n_1,n_2} \triangleq \frac{2}{n_1 n_2} \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \|X_i - Y_j\| - \frac{1}{n_1^2} \sum_{i=1}^{n_1} \sum_{k=1}^{n_1} \|X_i - X_k\| - \frac{1}{n_2^2} \sum_{j=1}^{n_2} \sum_{k=1}^{n_2} \|Y_j - Y_k\|, \tag{1.1}$$

where $\|X_i - Y_j\|, \|X_i - X_k\|, \|Y_j - Y_k\|$ are the distances from the two samples. Note that the statistic $\mathcal{E}_{n_1,n_2}$ solely depends on three types of inter-point distances: $\|X_i - Y_j\|$, $\|X_i - X_k\|$, $\|Y_j - Y_\ell\|$, $i, k = 1, \ldots, n_1$, $j, \ell = 1, \ldots, n_2$. Denote $m = n_1 + n_2$. The paper [41] have showed that it can be efficiently computed with computational complexity $O(m\log(m))$ in the univariate case (i.e., $p = 1$).

When X_i's and Y_j's are multivariate (i.e., we have $p > 1$), random projections have been proposed to find a fast approximation to the statistic $\mathcal{E}_{n_1,n_2}$. For example, [41] gave a fast algorithm that is based on random projections, which can achieve $O(nm \cdot \log(m))$ computational complexity, where n is the number of random projections. Note that the approach in [41] is a pure Monte Carlo approach. The recent advances in the quasi-Monte Carlo methods [53, 55] have demonstrated that in some settings, utilizing pre-determined projections can lead to better performance than the completely random ones in the pure Monte Carlo approach. Quasi-Monte Carlo methods sometimes enjoy faster rate of convergence, e.g., [54].

Our approach turns a distance calculation in a multivariate situation to the one in a univariate situation. The proposed approach

P1. first projects each multivariate variable along some pre-specified optimal directions to corresponding one-dimensional subspaces (the projected values are univariate),

P2. then the sum of the ℓ_1 norm of the projected values is used to approximate the

associated distance in the multivariate setting.

More specifically, let's suppose the multivariate variable is $v = (v_1, ..., v_p) \in \mathbb{R}^p$. Recall that the norm of v is

$$||v|| = \sqrt{\sum_{i=1}^{p} v_i^2}.$$

For $n \geq 1$, our objective is to identify the projection directions, which can be represented by vectors $u_1, u_2, ..., u_n \in \mathbb{R}^p$, and a predetermined constant $C_n \in \mathbb{R}$, such that for any $v \in \mathbb{R}^p$, we have

$$||v|| \approx C_n \sum_{i=1}^{n} \left| u_i^T v \right|. \tag{1.2}$$

Consequently in step **P2.**, when one needs to compute a distance $\|X_i - Y_j\|$, one can alternatively compute $C_n \sum_{i=1}^{n} \left| u_i^T X_i - u_i^T Y_j \right|$. Note that $u_i^T X_i$ and $u_i^T Y_j$ are univariate. Therefore the fast algorithm in the one-dimensional case can be utilized.

We continue with the example of the energy distances. Recall that the pre-specified directions are $u_1, ..., u_n$. The projected values of the corresponding multivariate variables then become

$$\begin{aligned} X_{wi} &= u_w^T X_i \in \mathbb{R}, w = 1, ..., n; i = 1, ..., n_1; \text{ and} \\ Y_{wj} &= u_w^T Y_j \in \mathbb{R}, w = 1, ..., n; j = 1, ..., n_2. \end{aligned}$$

The distance between any two multivariate variables can be approximated by the sum of these projections multiplying by a constant:

$$\|X_i - Y_j\| \approx C_n \sum_{w=1}^{n} |X_{wi} - Y_{wj}|.$$

Therefore, the statistic $\mathcal{E}_{n_1,n_2}$ in (1.1) can be approximated by

$$\begin{aligned}
\mathcal{E}_{n_1,n_2} &\approx C_n\Big(\frac{2}{n_1 n_2}\sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\sum_{w=1}^{n}\|X_{wi}-Y_{wj}\| - \frac{1}{n_1^2}\sum_{i=1}^{n_1}\sum_{k=1}^{n_1}\sum_{w=1}^{n}\|X_{wi}-X_{wk}\| \\
&\quad -\frac{1}{n_2^2}\sum_{j=1}^{n_2}\sum_{k=1}^{n_2}\sum_{w=1}^{n}\|Y_{wj}-Y_{wk}\|\Big) \\
&= C_n\Big(\frac{2}{n_1 n_2}\sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\sum_{w=1}^{n}|X_{wi}-Y_{wj}| - \frac{1}{n_1^2}\sum_{i=1}^{n_1}\sum_{k=1}^{n_1}\sum_{w=1}^{n}|X_{wi}-X_{wk}| \\
&\quad -\frac{1}{n_2^2}\sum_{j=1}^{n_2}\sum_{k=1}^{n_2}\sum_{w=1}^{n}|Y_{wj}-Y_{wk}|\Big).
\end{aligned} \tag{1.3}$$

The second equation is true because in the one-dimensional case, the ℓ_2 norm becomes the absolute value. Then one can apply the fast algorithms for univariate variables to calculate the energy statistic in (1.3).

Remark: *Our method is not restricted to the calculation of the energy statistic, or other distance-based statistics. It can also be applied to the calculation of the distance-based smooth kernel functions.*

In this chapter, we first give a detailed description of our strategy to find the optimal pre-specified projection directions. We formulate the searching for optimal projection directions problem as a minimax optimization problem. Let $\{u_1, u_2, \cdots, u_n\}$ denote the optimal set of projection directions, they should minimize the worst-case difference between the true distance and the approximate distance. Equation (1.4) below shows this idea in the mathematical form:

$$\min_{\substack{C_n,u_i:\\ \|u_i\|=1,i=1,\cdots,n}} \max_{v:\|v\|_2=\|X_i-Y_j\|} \left| C_n\sum_{w=1}^{n}\left|u_w^T(X_i-Y_j)\right| - \|X_i-Y_j\|\right|. \tag{1.4}$$

Discussion on how to solve the above problem is presented in Section 2.2.

In general, the problem in (1.4) is a nonconvex optimization problem, which is potentially NP-hard. We found that in two special cases, the optimal directions can be derived

analytically: (a) the 2-dimensional case and (b) when the dimension is equal to the number of projections. More details on these two special cases are presented in Section 2.3. In general cases, we propose a greedy algorithm to find the projection directions. Note that the greedy algorithm terminates at a local optimal solution to (1.4). In this case, we cannot theoretically guaranteed that the found directions correspond to the global solution to the problem in (1.4), which is the case in most nonconvex optimization problems. At the same time, the simulations show that our approach can still outperform the pure Monte Carlo approach in many occasions.

The rest of this chapter is organized as follows. Section 2.2 shows the formulation of our problem. Section 2.3 provides the analytical solutions to the problem in (1.4) in two special cases. Section 2.4 presents the numerical algorithm for the general cases. In Section 2.5, the simulation results of our method are furnished. Section 2.6 contains the conclusion and a summary of our work. All the technical proofs are relegated to the appendix (Section **??**).

We adopt the following notations. Throughout this chapter, we use p to denote the dimension of the data. The sample size is denoted by m. The number of projections is denoted by n.

2.2 Problem formulation

As mentioned above, in order to estimate the distance between two multivariate variables, we project them onto some pre-specified one-dimensional linear subspaces. We present details in the following. Suppose the multivariate variable is $v = (v_1, ..., v_p) \in \mathbb{R}^p$. Recall that the norm of vector v is

$$||v|| = \sqrt{\sum_{i=1}^{p} v_i^2}.$$

Our objective is to design $u_1, u_2, ..., u_n \in \mathbb{R}^p$, for $n \geq 1$, and $C_n \in \mathbb{R}$, such that for any $v \in \mathbb{R}^p$, we have

$$||v|| \approx C_n \sum_{i=1}^{n} \left| u_i^T v \right|. \tag{2.5}$$

We would like to turn a distance (i.e., norm) of a multivariate variable v into a weighted sum of the absolute values of some of its one dimensional projections (i.e., $u_i^T v$'s), knowing that the one dimensional projections may facilitate efficient numerical algorithms.

Without loss of generality, we may assume $||v|| = 1$. The approximation problem in (2.5) can be formulated into the following problem:

$$\min_{C_n, u_1, ..., u_n} \max_{v: ||v||_2 = 1} \left| C_n \sum_{i=1}^{n} \left| u_i^T v \right| - 1 \right|. \tag{2.6}$$

In words, we would like to select $u_1, ..., u_n$ and C_n such that the approximation in (2.5) has the minimal discrepancy in the worst case. One can verify that the problem in (2.6) and the problem in (1.4) share the same solutions.

To solve the problem in (2.6), the following two quantities are needed. For fixed $u_1, u_2, ..., u_n$, we define

$$V_{\max} = \max_{v: ||v||_2 = 1} \sum_{i=1}^{n} \left| u_i^T v \right|, \tag{2.7}$$

$$V_{\min} = \min_{v: ||v||_2 = 1} \sum_{i=1}^{n} \left| u_i^T v \right|, \tag{2.8}$$

where $V_{\max}$ and $V_{\min}$ are the maximum and minimum of $\sum_{i=1}^{n} \left| u_i^T v \right|$ among all possible v under the constraint $||v||_2 = 1$, respectively. With these two quantities (i.e., $V_{\max}$ and $V_{\min}$), we have the following result.

Theorem 2.2.1. *For given $u_1, u_2, ..., u_n \in \mathbb{R}^p$, the optimal value for C_n in the problem* (2.6) *is*

$$C_n = \frac{2}{V_{\min} + V_{\max}}.$$

Furthermore, the solutions of $u_1, u_2, ..., u_n$ *in problem* (2.6) *are identical to the solutions to the following problem:*

$$\max_{\substack{u_1,...,u_n: \\ ||u_i||=1, \forall i, 1\leq i\leq n}} \frac{V_{\min}}{V_{\max}}. \tag{2.9}$$

The above theorem indicates that the minimax problem in (2.6) is equivalent to the maximization problem in (2.9). Note that in general, both problems are nonconvex, therefore potentially NP-hard. In our analysis, we found that both formulations (in (2.6) and (2.9)) are convenient in various steps of derivation. Both of them are used in later analysis.

2.3 Derivable analytical results

We present the two special cases where analytical solutions are derivable. When the dimension is 2 (i.e., $p = 2$), we show in Section 2.3.1 that an analytical solution to the problem in (2.9) is available. In Section 2.3.2, we present another case (when the dimension of the data is equal to the number of projections, that is we have $n = p$) where an analytic solution to the problem in (2.9) is derivable.

2.3.1 Special case when the dimension is 2

When the multivariate variables are two-dimensional, we can get the exact optimal projections that minimize the worse-case discrepancy. The following theorem describes such a result.

Theorem 2.3.1. *When* $p = 2$*, the* 2*-dimensional vectors* $u_1, u_2, ..., u_n$ *can be represented by*

$$u_i = e^{\sqrt{-1}\theta_i}, i = 1, ..., n.$$

The optimal solution in (2.9) *has the form*

$$\theta_i = \frac{(i-1)\pi}{n} + k_i\pi, i = 1, ..., n \tag{3.10}$$

where each $k_i \in \mathbb{N}$.

Specially, when n is odd, the optimal solutions can be represented by the equally spaced points on the circle. Furthermore, we can get the error rate in the 2-dimensional case, as in the following theorem.

Theorem 2.3.2. *If* $u_1, \cdots, u_n$ *are chosen according to Theorem 2.3.1, we have*

$$\mathop{\mathbb{E}}_{v \sim \textit{Unif}(S^1)} \left\{ \left| C_n \sum_{i=1}^{n} \left| u_i^T v \right| - 1 \right|^2 \right\} = O\left(\frac{1}{n^2} \right).$$

Remark:*Theorem 2.3.2 can be used as a guidance of choosing the number of directions. Assume we would like to control the squared error to be* ϵ*. Then, we can get* $\frac{1}{n^2} = \epsilon$*, and therefore the number of directions should be larger than* $\frac{1}{\sqrt{\epsilon}}$.

In the above theorem, the random vector v is sampled independently from the Uniform distribution on the unit circle S^1. Note that the squared error rate is $O(1/n^2)$. The following theorem presents the corresponding rate for the pure random projections.

Theorem 2.3.3. *If* $u_1, \cdots, u_n$ *are selected base on Monte Carlo, we have*

$$\mathop{\mathbb{E}}_{u_i, v \sim \textit{Unif}(S^1)} \left\{ \left| C_n \sum_{i=1}^{n} \left| u_i^T v \right| - 1 \right|^2 \right\} = O\left(\frac{1}{n} \right).$$

In the above theorem, both random vector v and vectors u_i's are independently sampled from the Uniform distribution on the unit circle (S^1). The squared error rate in the pure Monte Carlo case is $O(1/n)$. These two theorems illustrate the theoretical advantage of adopting the pre-calculated projection directions (in relative to the random projections). Such a phenomenon has been discovered in the literature regarding the quasi-Monte Carlo methodology.

2.3.2 Second special case with provable result

When the dimension is larger than 2, the problem in (2.6) is challenging. There is some potentially relevant literature in mathematics, such as the searching for algorithms to locate the equally-distributed points on the surfaces of some high-dimensional spheres [42, 43, 44]. We fail to locate the exact solutions to our problem.

Our analysis indicates that when the number of projections is equal to the dimension, an analytical solution to the problem in (2.6) is derivable. We present details in the following. To derive our analytical solution in a special case, we need to revisit two quantities, $V_{\min}$ and $V_{\max}$, which have been introduced in (2.7) and (2.8). The following lemma is about $V_{\max}$.

Lemma 2.3.4. *For fixed $u_1, u_2, ..., u_n \in \mathbb{R}^p$, we have*

$$V_{\max} = \max_{s_i \in \{1,-1\}} \left\| \sum_{i=1}^{n} s_i u_i \right\|. \tag{3.11}$$

Lemma 2.3.4 points out a way to calculate $V_{\max}$, that is, given binary s_i's, finding out the linear combination $\sum_{i=1}^{n} s_i u_i$ with the maximal norm out of the all possible 2^n linear combinations. Let $\{s_i^{\max} \in \{1, -1\} : i = 1, ..., n\}$ denote the solution for (3.11) when $u_1, \cdots, u_n$ are given. The Algorithm 4 formally presents the aforementioned approach. Assume we are in the k-th loop, where the u_j's are known, which are denoted by $u_1^{(k)}, u_2^{(k)}, ..., u_n^{(k)}$. Let $s_i^{(k)}$'s denote the s_i's that can achieve $V_{\max}$ in the k-th loop. We have the Algorithm 4.

As for $V_{\min}$, suppose $v_{\min}$ is a minimizer of $V_{\min}$. We have the following property for $v_{\min}$.

Lemma 2.3.5. *For fixed $u_1, u_2, ..., u_n \in \mathbb{R}^p$, if Ω is an intersection of S^{p-1} and a linear subspace with at least 2 dimensions, then the solution to the minimization problem*

$$\min_{v \in \Omega} f(v) = \sum_{i=1}^{n} \left| u_i^T v \right|$$

Algorithm 4 Find $s_i^{\max}$'s in the k-loop

Initialization: Unit vectors $u_1^{(k)}, u_2^{(k)}, \ldots, u_n^{(k)} \in S^{p-1}$ are given.
Iteration: $s_i^{(k)}$'s.

1: **for** binary combination of $s_i^{(k)}$'s **do**
2: Calculate the value $\left\| \sum_{i=1}^{n} s_i u_i \right\|$.
3: **end for**
4: The binary combination that can make the value of $\left\| \sum_{i=1}^{n} s_i u_i \right\|$ be the maximum among all the possible values, is the $s_i^{\max}$'s, which is denoted as $s_i^{(k)}$'s.

must have $u_j^T v_{\min} = 0$ *for at least one* j $(1 \leq j \leq n)$.

Geometrically, the above lemma indicates that vector $v_{\min}$ should be orthogonal to at least one of the projection vector u_j. For vector $v_{\min}$, we will need the following definition to further our derivation.

Definition 2.3.6 (maximal subset)**.** *We call* $\Omega(v_{\min})$ *a maximal subset of the set* $\{u_1, ..., u_n\}$ *if it satisfies*

$$\Omega(v_{\min}) = \left\{u_j : u_j^T v_{\min} = 0\right\} \subset \{u_1, ..., u_n\},$$

and it cannot be a strict subset for another $\Omega(v'_{\min})$ *where* $v'_{\min}$ *is a minimizer that is different from* $v_{\min}$.

Lemma 2.3.5 ensures that the set $\Omega(v_{\min})$ cannot be empty. The following lemma shows that the linear subspace that is spanned by the elements of $\Omega(v_{\min})$ must have certain dimensions.

Lemma 2.3.7. *If* $\Omega(v_{\min})$ *is a maximal subset of* $u_1, ..., u_n$*, we must have*

$$\textit{rank}\left(\Omega(v_{\min})\right) = p - 1,$$

for any minimizer $v_{\min}$.

Recall p is the dimension of the data. The above lemma essentially states that the space

that is spanned by the elements of $\Omega(v_{\min})$ is the orthogonal complement subspace of the one-dimensional space that is spanned by the vector $v_{\min}$.

One direct corollary of Lemma 2.3.7 is that the cardinality of the set $\Omega(v_{\min})$ is at least $p-1$. Consequently, the total number of possible sets (of $\Omega(v_{\min})$) is no more than $\binom{n}{p-1}$. This inspires us to use Algorithm 5 to find $v_{\min}$ as well as $\Omega(v_{\min})$ if all the u_j's are given. Here suppose we are in the k-th loop where the u_j's are known, which are $u_1^{(k)}, u_2^{(k)}, ..., u_n^{(k)}$.

Algorithm 5 Find $v_{\min}$ and $\Omega(v_{\min})$ in the k-loop

Initialization: Unit vectors $u_1^{(k)}, u_2^{(k)}, \ldots, u_n^{(k)} \in S^{p-1}$ are given.
Iteration: $v^{(k)}$ and $\Omega(v^{(k)})$.

1: **for** $(p-1)$ combination of $u_i^{(k)}$'s, denoted as S_t^u **do**
2: **while** rank$(S_t^u) < p-1$ **do**
3: Add another u_j that is not in the set S_t^u;
4: **end while**
5: Find the orthogonal direction of the set S_t^u, which is one of the candidates of $v^{(k)}$, denoted as $v_t^{(k)}$, and calculate the value of $f(v_t^{(k)}) = \sum_{i=1}^{n} \left| \left(u_i^{(k)}\right)^T v_t^{(k)} \right|$.
6: **end for**
7: The $v_t^{(k)}$, that can make the value of $f(v_t^{(k)})$ be the minimum among all the possible $f(v_t^{(k)})$ values, is the $v_{\min}$, which is denoted as $v^{(k)}$, and the corresponding S_t^u set is the set $\Omega(v_{\min})$, which is denoted as $\Omega(v^{(k)})$.

From Lemma 2.3.7 we can get the exact solution for the special case when the number of projection directions is equal to the dimension of the multivariate variables, which is described in the following theorem.

Theorem 2.3.8. *When the number of projections is equal to the dimension of the data, i.e., we have $n = p$, the optimal solution in* (2.9) *satisfies the following condition:*

$$u_i^T u_j = 0, \forall i \neq j. \tag{3.12}$$

The above is equivalent to stating that the set $\{u_1, u_2, \cdots, u_n\}$ forms an orthonormal basis in $\mathbb{R}^p$.

2.4 Numerical approach in general cases

When $p > 2$ and $n \neq p$, we propose an algorithm to identify the optimal projections $u_1, u_2, ..., u_n$, such that they solve (2.9). Per Lemma 2.3.4 and the definition of $s_i^{\max}$'s, the $V_{\max}$ can be written as:

$$V_{\max} = \left\| \sum_{i=1}^{n} s_i^{\max} u_i \right\|.$$

According to Lemma 2.3.7, we have

$$\begin{aligned} V_{\min} = \sum_{i=1}^{n} \left| u_i^T v_{\min} \right| &= \sum_{u_i \in \Omega(v_{\min})} \left| u_i^T v_{\min} \right| + \sum_{u_i \notin \Omega(v_{\min})} \left| u_i^T v_{\min} \right| \\ &= \sum_{u_i \notin \Omega(v_{\min})} \left| u_i^T v_{\min} \right|. \end{aligned}$$

So when $u_1, \cdots, u_n$ are given, $\frac{V_{\min}}{V_{\max}}$ can be written as

$$\frac{V_{\min}}{V_{\max}} = \frac{\sum\limits_{u_i \notin \Omega(v_{\min})} \left| u_i^T v_{\min} \right|}{\left\| \sum\limits_{i=1}^{n} s_i^{\max} u_i \right\|}, \tag{4.13}$$

where $v_{\min}$ and $\Omega(v_{\min})$ are defined in Section 2.3.2. We assume that the set $\Omega(v_{\min})$ corresponds to the minimum over all $\binom{n}{p-1}$ possible sets, and $(s_i^{\max})$'s maximize the norm of $\sum\limits_{i=1}^{n} s_i^{\max} u_i$.

We use a method that is similar to the coordinate descent algorithm [51, 45] to search for the optimal solutions of (2.9). Details of our algorithm can be found in Algorithm 6. The optimal solution can be achieved by in a loop, maximizing (4.13) with respect to one u_i, while the others are fixed. We then iteratively maximize the objective function in (4.13) until the value of the objective function (4.13) cannot be increased.

We derived the iteration strategy in the following. Let $v^{(k)}$ be the minimizer of $\sum\limits_{i=1}^{n} \left| u_i^T v \right|$ at the kth iteration. Let $\Omega^{(k)}$ denote the minimum over all $\binom{n}{p-1}$ possible sets at the kth iteration. For any $u_j^{(k)} \notin \Omega^{(k)}$, without loss of generality, we assume that $u_1 \notin \Omega^{(k)}$. The

objective function in (4.13) can be written as

$$\frac{V_{\min}}{V_{\max}} = \frac{\left|u_1^T v^{(k)}\right| + \sum\limits_{i>1, u_i \notin \Omega^{(k)}} \left|u_i^T v^{(k)}\right|}{\left\| s_1^{\max} u_1 + \sum\limits_{i=2}^{n} s_i^{\max} u_i \right\|}. \tag{4.14}$$

Without loss of generality, we can assume $s_1^{\max} = 1$. This is because, recalling that $(s_i^{\max})$'s are binary, we have

$$\left\| s_1^{\max} u_1 + \sum_{i=2}^{n} s_i^{\max} u_i \right\| = \left\| u_1 + \sum_{i=2}^{n} s_1^{\max} s_i^{\max} u_i \right\|.$$

The expression in (4.14) can then be rewritten as

$$\frac{\left|u_1^T v^{(k)}\right| + A}{\|u_1 + B\|}, \tag{4.15}$$

where

$$A = \sum_{i>1, u_i \notin \Omega^{(k)}} \left|u_i^T v^{(k)}\right|, \text{ and } B = \sum_{i=2}^{n} s_i^{\max} u_i.$$

Note that quantities A and B do not depend on u_1. Our objective is to derive a strategy to maximize the quantity in (4.15) as a function of the vector variable u_1.

We first solve a constrained version of the above maximization problem. We define $\Sigma(v, \theta) = \{x : \|x\| = 1, \langle x, v \rangle = \theta\}$, for any fixed $\theta \in [0, \pi)$, where $\langle \cdot, \cdot \rangle$ denote the angle between two vectors. Conditioning on $u_1 \in \Sigma(v, \theta)$, and $v = v^{(k)}$, maximizing the function in (4.15) is equivalent to maximizing the following function:

$$\frac{|\cos\theta| + A}{\|u_1 + B\|}. \tag{4.16}$$

Note that the numerator is not a function of u_1. Consequently, it is equivalent to minimizing

$$\|x + B\|, \text{ where } x \in \Sigma(v, \theta).$$

The following lemma presents an analytical solution to the above minimization problem.

Lemma 2.4.1. *Given a vector B, a constant $\theta \in [0, \pi)$, and a unit-norm vector v, the solution to the following problem*

$$\min_{x:\|x\|=1,\langle x,v\rangle=\theta} \|x + B\|^2 \tag{4.17}$$

is

$$x = v\cos\theta + \frac{|\sin\theta|}{\sqrt{B^TB - (v^TB)^2}}\left[(v^TB)v - B\right]. \tag{4.18}$$

Using the solution in (4.18) to substitute the u_1 in (4.16), we have

$$\frac{|\cos\theta| + A}{\|u_1 + B\|} = \frac{|\cos\theta| + A}{\left\| v\cos\theta + B + \frac{|\sin\theta|}{\sqrt{B^TB-(v^TB)^2}}\left[(v^TB)v - B\right]\right\|}. \tag{4.19}$$

Maximizing (4.16) with respect to θ is equivalent to maximizing (4.19). For fixed A, B, and v, the right hand side of (4.19) is a function of θ. The following Theorem 2.4.2 gives the solution to the above problem.

Theorem 2.4.2. *The solutions of maximizing* (4.16) *with respect to θ are the zeros of the following function:*

$$g(\theta) = \begin{cases} \sqrt{B^TB}\left[\cos\alpha + A\cos(\alpha-\theta) - \sin\theta\sin(\alpha-\theta)\right] & \\ -(1 + B^TB)\sin\theta, & \text{if } \theta \in [0, \frac{\pi}{2}), \\ \sqrt{B^TB}\left[-\cos\alpha + A\cos(\alpha-\theta) + \sin\theta\sin(\alpha-\theta)\right] & \\ +(1 + B^TB)\sin\theta & \text{if } \theta \in [\frac{\pi}{2}, \pi), \end{cases} \tag{4.20}$$

where α satisfies $\sin\alpha = \frac{v^TB}{\sqrt{B^TB}}$, *and* $\cos\alpha = \frac{\sqrt{B^TB-(v^TB)^2}}{\sqrt{B^TB}}$.

The above theorem indicates that one can adopt a line search algorithm to compute for θ.

Based on all the above, the Algorithm 6 (below) furnishes a coordinate ascent scheme to maximize the objective in (2.9).

Algorithm 6 Optimal projection algorithm

Initialization: Set a threshold $\Delta > 0$, initial unit vectors $u_1^{(0)}, u_2^{(0)}, ..., u_n^{(0)} \in S^{p-1}$. Thus, by Algorithm 4 and 5, we can get the corresponding values $v^{(0)}$,$\Omega^{(0)}(v^{(0)})$, and $s_i^{(0)}$'s.

repeat
  In the k-th loop, suppose the previous $u_1^{(k-1)}, u_2^{(k-1)}, \ldots, u_n^{(k-1)}$ are known.
  for $u_j^{(k-1)} \notin \Omega^{(k-1)}(v^{(k-1)})$ **do**
    Find the zeros of the function $g(\theta)$ in (4.20) in Theorem 2.4.2, where $v = v^{(k-1)}$, $B = \sum_{i \neq j} s_j^{(k-1)} s_i^{(k-1)} u_i^{(k-1)}$, and denote the zeros as θ^*.
    According to Lemma 2.4.1, the new $u_j^{(k)}$ would be $v \cos\theta^* + \frac{|\sin\theta^*|}{\sqrt{B^T B - (v^T B)^2}} \left[(v^T B)v - B\right]$.
    By Algorithm 4 and 5, we can get the corresponding values $v^{(k)}$,$\Omega^{(k)}(v^{(k)})$, and $s_i^{(k)}$'s, based on the newly updated u_j's, which also give us the value of $V_{\min}$ and $V_{\max}$.
    Compute $V_{\min}/V_{\max}$.
  end for
  Pick the $u_j^{(k)} \notin \Omega^{(k-1)}(v^{(k-1)})$ that gives the maximal value of $V_{\min}/V_{\max}$ in the above loop.
  if The value of $V_{\min}/V_{\max}$ decreases **then**
    Go back to $u_j^{(k-1)}$.
  end if
until The increment of $V_{\min}/V_{\max}$ is less than Δ.

2.5 Simulations

In the previous section, the optimal projections for both the special cases and the general case are provided. The simulations will follow the same order. The simulations are about the comparison of the Monte Carlo method and our method for the special cases and then for the general case.

According to [41], Monte Carlo method is to select some random directions, denoted as w_i, $i = 1, \ldots, n$, on the unit sphere S^{p-1} and project the vector we would like to estimate,

that is v, along these directions, so the norm of the vector v could be estimated as

$$\|v\| \approx C'_p \frac{1}{n} \sum_{i=1}^{n} |w_i^T v|,$$

where $C'_p = \frac{\sqrt{\pi}\Gamma(\frac{p+1}{2})}{\Gamma(\frac{p}{2})}$.

In all the experiments, we randomly select 100 unit vectors on the sphere as the vectors that we would like to estimate, in order to get the mean squared error for comparison between the Monte Carlo method and the method we propose.

2.5.1 When the dimension is 2

When the dimension is equal to 2, the exact solution can be found as well as the mean squared error rate. So we randomly select 100 unit vectors on the sphere as the vectors that we would like to estimate. For both the Monte Carlo method and our optimal projection method, we calculate the mean squared error over these 100 vectors. More specifically, the squared error between the true norm of the vector, which is 1, and the estimated norm is calculated for each of the 100 unit vectors when the number of directions is fixed. By taking the mean of the 100 squared errors from the previous step, we get the mean squared error for given number of directions. The number of directions used in our simulation is from 2 to 10000. Figure 2.1 shows the comparison between our method and Monte Carlo method regarding the logarithm of the mean squared error and the number of projection directions. From the figure, we can see that our method performs better than the Monte Carlo, and the advantage becomes more obvious when the number of projection directions increases.

2.5.2 When we have $n = p$

When the dimension p is equal to the number of projection directions n, recall that in Theorem 2.3.8, we give the exact solution of the pre-specified directions. Similar to what

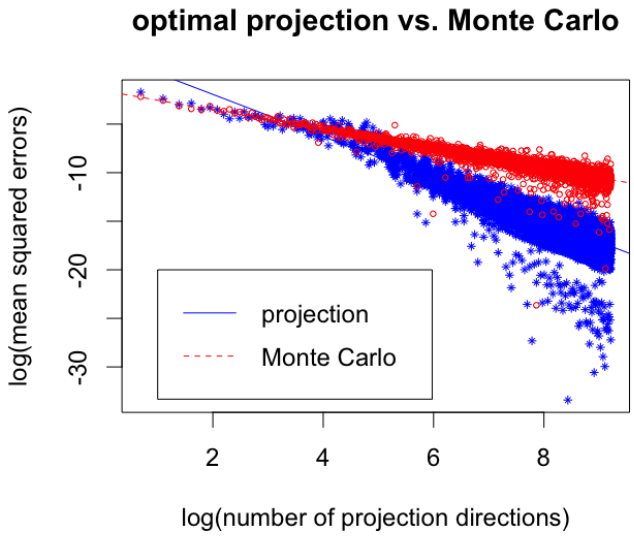


Figure 2.1: Optimal projection vs. Monte Carlo in the 2 dimensional case

we have done in the 2-dimensional case, we randomly select 100 unit vectors on the sphere S^{p-1}, with dimension p varying from 8 to 11. So the number of projection directions is varying from 8 to 11 correspondingly. We calculate the mean squared error of both the Monte Carlo method and our optimal projection method for each p using the same strategy as before. The details are in the Figure 2.2, where the x-axis represents the dimension, and y-axis represents the mean squared error.

2.5.3 General setting: $n \geq p$

When the dimension p is larger than 2 and $n \neq p$, the exact solution of (2.9) can not be obtained. Therefore, we adopt the Algorithm 6. Like in previous simulations, we randomly select 100 unit vectors on the sphere S^{p-1}, with dimension p varying from 3 to the number of directions minus 1, and the fixed number of directions to be 8, 9, 10, 11, respectively, and calculate the mean squared error of both the Monte Carlo method and our optimal projection method for each p using the same strategy as before. Figure 2.3, 2.4, 2.5 and 2.6 show the comparison, where the x-axis represents the dimension, and y-axis represents the mean squared error.

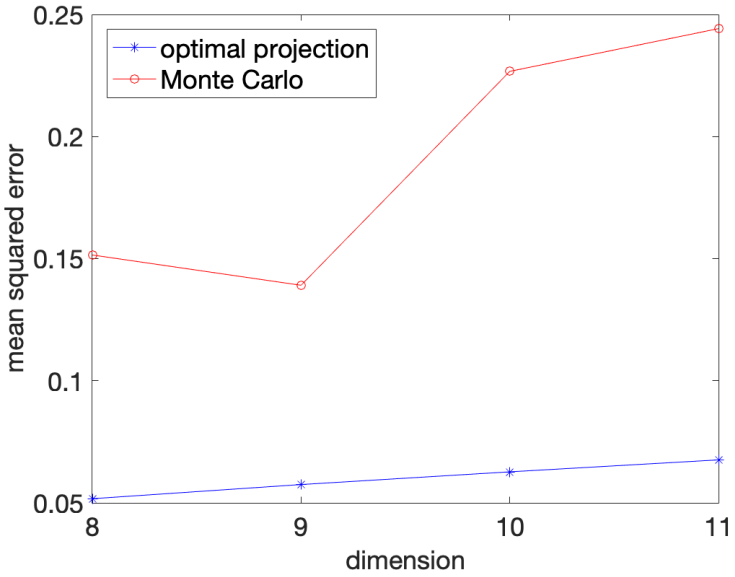


Figure 2.2: Optimal projection vs. Monte Carlo in the $n = p$ case

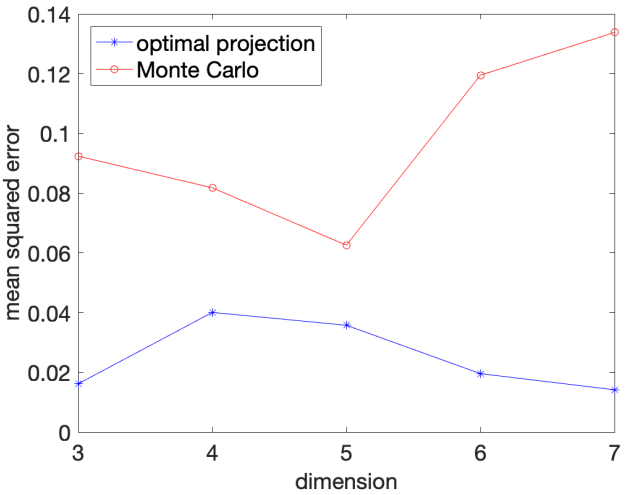


Figure 2.3: Optimal projection vs. Monte Carlo for dimension varying from 3 to 7 in the case $n = 8$

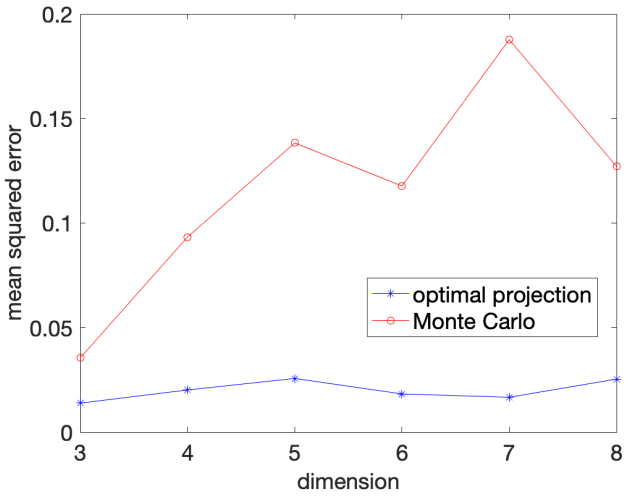


Figure 2.4: Optimal projection vs. Monte Carlo for dimension varying from 3 to 8 in the case $n = 9$

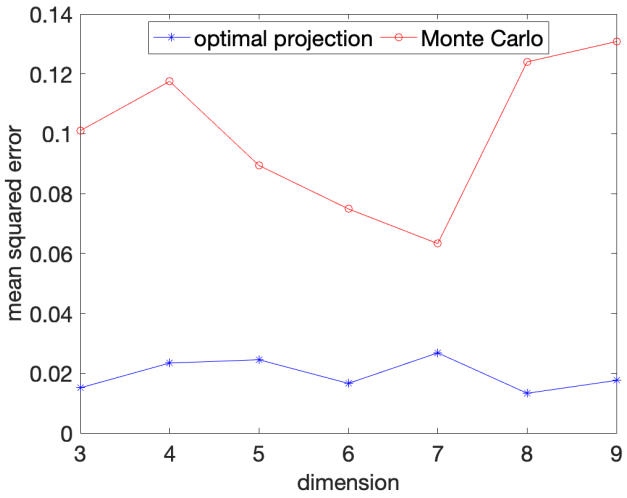


Figure 2.5: Optimal projection vs. Monte Carlo for dimension varying from 3 to 9 in the case $n = 10$

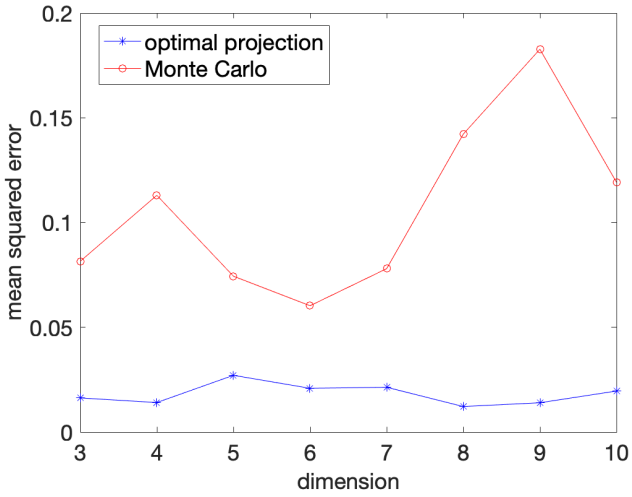


Figure 2.6: Optimal projection vs. Monte Carlo for dimension varying from 3 to 10 in the case $n = 11$

Overall, we can see that our method performs better than the Monte Carlo method.

2.6 Conclusion

We propose a new method to calculate the distance, which is critical in computing the distance-based statistics, and can also be utilized in the calculation of the kernel functions that are distance-based and smooth. The main idea is to use the sum of the norms of the projections along a set of pre-calculated directions to approximate the original norm. By doing so, one can utilize the fast algorithm for univariate variables that has been proposed by [40]. The advantage is that the computational complexity is reduced from $O(m^2)$ to $O(m\log(m))$ where m is the sample size. These pre-specified directions can be found by minimizing the difference between the estimated distance and the true value in the worst case. The associated problem is eventually a nonconvex optimization problem. We derive the exact solutions when dimension is equal to either 2 or the number of projection directions. In general cases, we propose an algorithm to find the projection directions. The simulations show the advantage of the proposed method versus the pure Monte Carlo approach, via comparing the mean squared errors.

CHAPTER 3
A NEW SEMIDEFINITE PROGRAMMING ALGORITHM FOR POWER FLOW AND POWER SYSTEM STATE ESTIMATION

This chapter proposes a new semidefinite programming algorithm to solve both the power flow and poweer system state estimation problem. Both two kinds of problems are non-convex, and convex relaxation is the typical approach to non-convexity in power systems area, while the objective functions are required to be carefully designed in order to keep the equivalency before and after relaxation. In this chapter, we first reformulate the two types of complex-valued problems as non-convex real-valued ones. Rather than convex relaxation, we further show that the solution of these problems can be found by alternately solving two semidefinite programming problems. Convergence analysis is provided, and we also give the conditions under which the equivalency holds between the original problem and the newly proposed sequence optimization problem. Numerical results on the typical bus systems demonstrate that our method is more applicable than the classical weighted least square method when the voltage angles are not close to zero. In addition, our method shows strong robustness regarding the start point and is significantly advantageous over the others especially when bad data exist.

3.1 Introduction

An electrical power system facilitates supplying, delivering, and consuming the electric power, where precise operating points and state estimation are important, not only for both the reliability and security of the whole power networks, but also for the economic considerations. In order to get accurate solutions, power flow analysis and power system state estimation have been developed, studied, and applied in many practical problems, such as optimal power flow, network reconfiguration, and so on [73]. The remaining of section 2.1

is organized as follows: subsection 3.1.1 gives a brief review of previous studies in power flow analysis and power system state estimation; subsection 3.1.2 states our contributions; subsection 3.1.3 includes the notations in this chapter.

3.1.1 Previous Studies

Power flow analysis is a steady state analysis to determine the voltages (both magnitudes and angles) for each bus in a power flow system given loading conditions, so as to get the whole picture of the power flows of the system. It is a helpful and necessary step for the future planning. For example, power flow analysis can be performed for the study of various hypothetical situations, such as, a line taken off for repairing, to help people better prepare for the potential system failures. The basic equations in power flow analysis are a set of quadratic equations derived from the laws in physics. In general, it is a strong NP-hard problem [56].

Many methods have been developed in order to solve the power flow equations [57]. One direction is to find the solutions by the exact expressions of the power flow equations, the other is to get the approximate solutions to the power flow problem [58]. For the latter one, the DC power flow equations are derived ([59, 60]) for simplification, such that fast algorithms can be implemented. At the same time, however, accuracy can only be achieved to a reasonable degree. Therefore there is a trade-off between speed and accuracy. This chapter focuses on the former direction, to explore methods of solving the AC equations with the emphasis on accuracy.

To solve the power flow equations, the Gauss-Seidel method and the Newton-Raphson method [61], and their variants [62, 63] are the most popular iterative methods in power flow analysis, in which the quadratic convergence is guaranteed as long as the initializations are reasonable. This type of methods have been used for many years and the disadvantages that have been criticized are mainly regarding the heavy reliance on the starting values (the convergence guarantee can only be proved if the starting point is close to the true

solution). In addition, these methods are only applicable when voltage phases are close to zero. Another way of solving it is to formulate this problem as a semidefinite programming by convex relaxation ([78, 77]). Lots of recent work have been devoted in this range, such as [65, 66, 67, 68].

Besides power flow analysis, a highly related topic, state estimation problem is also common in practice. The purpose of state estimation is to get the voltages estimation (both magnitudes and angles) for each bus given system measurements [69], Weighted least squares estimation has been widely used in state estimation, but its sensitivity to outliers makes it not robust. Therefore, least absolute value estimation has been implemented in order to get robustness. No matter weighed least squares or least absolute value method, linear programming is a common strategy to solve the nonconvex problem [70, 71], which intrigues the linear approximation in each iteration step. Similar to power flow analysis, semidefinite programming can also be utilized on robust state estimation, which is usually done by relax the nonconvex constraints to convex ones, in order to be solvable.

Overall, whether in power flow analysis or robust state estimation, the formulated problem is indeed to find a rank one matrix with a set of semidefinite constraints. Current literatures use different convex relaxation strategies to solve it: Weng et al. drop the rank one constraint and do weighted least square (WLS) or weighted least absolute value (WLAV) estimation in [64] ; In the chapter [67] and [72], the authors relax the rank one problem as a trace minimization problem by carefully choosing the multiplied matrix in the objective trace function; other techniques, like stochastic gradient in [68], are also applied. Convex relaxation, however, needs to be handled delicately as the accuracy is garanteed only if relaxation conditions are satisfied. Because of the potential intractability of these relaxation methods in practice, this chapter targets at the original nonconvex problem and gets the solution by alternately solving a pair of convex optimization problems.

3.1.2 Our Contributions

Our method can be seen as an improvement of the conic relaxation method in [67]. We follow the same structure to transform the net power and branch power constraints into the trace formulation by constructing a series of matrices based on the admittance matrices of both the net, the from-end branches and the to-end branches. Instead of constructing the problem on complex domain as in [67], we prove that the problem can be equivalently formulated into a complex-valued problems with a real-valued objective function.

Furthermore, while most existing literatures including [67] make use of convex relaxation to overcome the nonconvexity of the original problem, ours does not, in consideration of the potential impracticability of the relaxing requirements. For example, in [67], a preconstructed matrix $\boldsymbol{M}_0$ is added as an additional constraint in the relaxed convex problem, which depends on the value of the solution to the problem. More specifically, as stated in [67], "it is impossible to design the matrix $\boldsymbol{M}_0$ in advance without knowing the phase angle difference $\measuredangle v_s - \measuredangle v_t$", where $\measuredangle v_s$ and $\measuredangle v_t$ are the voltage angles of the s-th and t-th bus respectively. Although they give several options to construct the matrix $\boldsymbol{M}_0$ based on susceptance and conductance matrices, one needs to choose carefully to keep the relaxation equivalency, which brings uncertainty of the accuracy of the solutions in practice. Moreover, the chapter [67] assumes that we have all the measurements of voltage magnitudes for each bus, which may not be practical in reality either. In order to overcome the potential difficulties, we handle the problem differently. Rather than constructing matrix $\boldsymbol{M}_0$ to relax the nonconvex optimization problem into a convex one, we formulate it as a sequence optimization problem which iteratively solves two convex problems, with no need to construct additional matrices and can be implemented in more general cases. We show that this strategy can not only be used in the classical power flow problem, but also be extended to the power system state estimation problem with strong robustness. Moreover, [67] is mainly focused on the scenarios where only the measurements of nodal voltage magnitudes and branch active power flows are given, whereas our method is applicable in more

general cases. Besides, in comparison with weighted least squares method, our method can perform well when true voltage angles are large while weighted least squares can produce reasonable solutions when true voltage angles are close to zero.

3.1.3 Notations

In this chapter, the boldface lowercase letters represent vectors, and the boldface uppercase letters represent matrices. Sets are represented by calligraphic letters. More specifically, $\mathbb{H}^N$ represents the set of $N \times N$ Hermitian matrices, $\mathbb{R}^N$ represents the set of $N \times 1$ vectors, and $\mathbb{S}^{2N}$ represents the set of $2N \times 2N$ symmetric matrices. The symbol $(\cdot)^T, (\cdot)^*$ represents the transpose and the conjugate transpose of a vector or a matrix. The symbol $\text{tr}(\cdot), \mathcal{R}\cdot, \mathcal{I}\cdot,$ and $\text{rank}(\cdot)$ represent the trace, the real part, the imaginary part, and the rank of a matrix. $\text{diag}(\cdot)$ denotes a diagonal matrix whose diagonal entries are given by the vector inside the parentheses. $\boldsymbol{X} \succeq 0$ indicates that the matrix $\boldsymbol{X}$ is nonnegative semidefinite.

3.2 Preliminaries

Consider an electric power network with N buses and L branches. The set $\mathcal{N} = \{1, \cdots, N\}$ is denoted as the set of buses, and the set $\mathcal{L} = \{1, \cdots, L\}$ is denoted as the set of branches. Vector $\boldsymbol{v} = (v_k)_{N\times 1} \in \mathbb{C}^{N\times 1}$ represents the nodal complex voltage vector. The net injected complex power at bus k ($k \in \mathcal{N}$) is denoted as $s_k = p_k + q_k j$, where $j = \sqrt{-1}$, and the complex power injections entering the line $l \in \mathcal{L}$ through the from and to ends of the branch as $s_{lf} = p_{lf} + q_{lf} j$ and $s_{lt} = p_{lt} + q_{lt} j$, respectively. $\boldsymbol{Y}$ is denoted as the nodal admittance matrix of the power network, and $\boldsymbol{Y}_f$ and $\boldsymbol{Y}_t$ are the from and to branch admittance matrices respectively. From Ohm's law, the overall current, and the from and to branch current can be written as

$$\boldsymbol{i} = \boldsymbol{Y}\boldsymbol{v}, \quad \boldsymbol{i}_f = \boldsymbol{Y}_f\boldsymbol{v}, \quad \boldsymbol{i}_t = \boldsymbol{Y}_t\boldsymbol{v}.$$

Let $\{\boldsymbol{e}_1, \cdots, \boldsymbol{e}_N\}$ and $\{\boldsymbol{d}_1, \cdots, \boldsymbol{d}_L\}$ denote the canonical basis vectors in $\mathbb{R}^N$, and the canonical basis vectors in $\mathbb{R}^L$, respectively. We define

$$\begin{aligned} \boldsymbol{E}_k &= \boldsymbol{e}_k \boldsymbol{e}_k^T, \\ \boldsymbol{Y}_{k,p} &= \frac{1}{2}\left(\boldsymbol{Y}^* \boldsymbol{E}_k + \boldsymbol{E}_k \boldsymbol{Y}\right), \\ \boldsymbol{Y}_{k,q} &= \frac{j}{2}\left(\boldsymbol{E}_k \boldsymbol{Y} - \boldsymbol{Y}^* \boldsymbol{E}_k\right). \end{aligned} \tag{2.1}$$

As we know $s_k = i_k^* v_k$, then for each $k \in \mathcal{N}$, the voltage magnitude $|v_k|$, and the active and reactive net injected power p_k and q_k can be expressed as

$$\begin{aligned} |v_k|^2 &= \text{tr}\left(\boldsymbol{E}_k \boldsymbol{v}\boldsymbol{v}^*\right), \\ p_k &= \text{tr}\left(\boldsymbol{Y}_{k,p} \boldsymbol{v}\boldsymbol{v}^*\right), \\ q_k &= \text{tr}\left(\boldsymbol{Y}_{k,q} \boldsymbol{v}\boldsymbol{v}^*\right). \end{aligned} \tag{2.2}$$

Similarly, on the lth branch from node i to node j, if we define

$$\begin{aligned} \boldsymbol{Y}_{l,p_f} &= \frac{1}{2}\left(\boldsymbol{Y}_f^* \boldsymbol{d}_l \boldsymbol{e}_i^T + \boldsymbol{e}_i \boldsymbol{d}_l^T \boldsymbol{Y}_f\right), \\ \boldsymbol{Y}_{l,p_t} &= \frac{1}{2}\left(\boldsymbol{Y}_t^* \boldsymbol{d}_l \boldsymbol{e}_j^T + \boldsymbol{e}_j \boldsymbol{d}_l^T \boldsymbol{Y}_t\right), \\ \boldsymbol{Y}_{l,q_f} &= \frac{j}{2}\left(\boldsymbol{e}_i \boldsymbol{d}_l^T \boldsymbol{Y}_f - \boldsymbol{Y}_f^* \boldsymbol{d}_l \boldsymbol{e}_i^T\right), \\ \boldsymbol{Y}_{l,q_t} &= \frac{j}{2}\left(\boldsymbol{e}_j \boldsymbol{d}_l^T \boldsymbol{Y}_t - \boldsymbol{Y}_t^* \boldsymbol{d}_l \boldsymbol{e}_j^T\right), \end{aligned} \tag{2.3}$$

then, for each line $l \in \mathcal{L}$ from node i to node j, $p_{l,f}, q_{l,f}$ – the active and reactive power of the from end of the branch l, and $p_{l,t}, q_{l,t}$ – the active and reactive power of the to end of the branch l can be expressed as

$$p_{l,f} = \text{tr}\left(\boldsymbol{Y}_{l,p_f} \boldsymbol{v}\boldsymbol{v}^*\right), \quad q_{l,f} = \text{tr}\left(\boldsymbol{Y}_{l,q_f} \boldsymbol{v}\boldsymbol{v}^*\right),$$
$$p_{l,t} = \text{tr}\left(\boldsymbol{Y}_{l,p_t} \boldsymbol{v}\boldsymbol{v}^*\right), \quad q_{l,t} = \text{tr}\left(\boldsymbol{Y}_{l,q_t} \boldsymbol{v}\boldsymbol{v}^*\right).$$

Therefore, if we assume there are M measurements, and the set $\mathcal{M}$ is defined as $\mathcal{M} = \{1, \cdots, M\}$, then the nodal and line measurements can be represented by the following quadratic formulations of the complex voltage $\boldsymbol{v}$

$$\text{tr}\,(\boldsymbol{M}_j \boldsymbol{v}\boldsymbol{v}^*) = z_j, \;\; \forall j \in \mathcal{M},$$

where $\{\boldsymbol{M}_j\}_{j\in\mathcal{M}}$ can be arbitrary measurement matrices from (2.1) and (2.3), and $\{z_j\}_{j\in\mathcal{M}}$ are the known nodal or line measurements.

Let $\boldsymbol{X} = \boldsymbol{v}\boldsymbol{v}^*$. Then, the power flow problem can be written as

$$\begin{aligned} \text{find} \quad & \boldsymbol{v} \in \mathbb{C}^N \\ \text{s.t.} \quad & \text{tr}\,(\boldsymbol{M}_j \boldsymbol{X}) = z_j, \;\; \forall j \in \mathcal{M}, \\ & \boldsymbol{X} = \boldsymbol{v}\boldsymbol{v}^*. \end{aligned} \tag{2.4}$$

3.3 Power Flow Analysis

We know from linear algebra that $\boldsymbol{X} = \boldsymbol{v}\boldsymbol{v}^*$ is equivalent to $\boldsymbol{X} \succeq 0$ and $\text{rank}(\boldsymbol{X}) = 1$. Therefore the problem (2.4) is equivalent to the following problem

$$\begin{aligned} \text{find} \quad & \boldsymbol{X} \in \mathbb{H}^N \\ \text{s.t.} \quad & \text{tr}\,(\boldsymbol{M}_j \boldsymbol{X}) = z_j, \;\; \forall j \in \mathcal{M}, \\ & \text{rank}(\boldsymbol{X}) = 1, \\ & \boldsymbol{X} \succeq 0. \end{aligned} \tag{3.5}$$

So our focus now is to solve problem (3.5), which is the equivalent version of problem (2.4). Recall that $\boldsymbol{X}$ is a complex matrix. We denote the real and image part of $\boldsymbol{X}$ as $\mathcal{R}\boldsymbol{X}$

and $\mathcal{I}\boldsymbol{X}$, respectively, and let the real matrix $\boldsymbol{Z}$ be

$$\boldsymbol{Z} = \begin{bmatrix} \mathcal{R}\boldsymbol{X} & -\mathcal{I}\boldsymbol{X} \\ \mathcal{I}\boldsymbol{X} & \mathcal{R}\boldsymbol{X} \end{bmatrix}. \tag{3.6}$$

is transformed to real domain as shown in Proposition 3.3.1 shows further reformulation of Problem (2.4) based on Problem (3.5).

Proposition 3.3.1. *Then problem* (2.4) *is equivalent to*

$$\begin{aligned} \textit{find} \quad & \boldsymbol{X} \in \mathbb{H}^N \\ \textit{s.t.} \quad & \textit{tr}\,(\boldsymbol{M}_j\boldsymbol{X}) = z_j, \ \ \forall j \in \mathcal{M}, \\ & \textit{rank}(\boldsymbol{Z}) = 2, \\ & \boldsymbol{X} \succeq 0. \end{aligned} \tag{3.7}$$

From the above we can see that as long as we can get the solution of the problem (3.7), the power flow analysis problem can be solved. The difficult part is that problem (3.7) is a nonconvex NP-hard optimization problem. According to the chapter in [74, Chapter 4.5], the problem can be expressed as a sequence optimization problem. Suppose the eigenvalues of $\boldsymbol{Z} \in \mathbb{S}^{2N}$ are $\lambda_1(\boldsymbol{Z}) \geq \lambda_2(\boldsymbol{Z}) \geq \ldots \lambda_{2N}(\boldsymbol{Z})$. If $\boldsymbol{Z}$ satisfies the following conditions in the problem (3.7): $\text{rank}(\boldsymbol{Z}) = 2,$ and $\boldsymbol{Z} \succeq 0$, the sum $\sum_{i=3}^{2N} \lambda_i(\boldsymbol{Z})$ should reach its minimum 0. Moreover, since matrix $\boldsymbol{X}$ satisfies the linear constraints, the rank of $\boldsymbol{X}$ cannot be smaller than 1, which indicates the rank of $\boldsymbol{Z}$ cannot be smaller than 2 based on the proof of Proposition 3.7. Therefore, if the problem (3.7) is feasible, then there exists some matrix $\boldsymbol{Z}$ such that $\text{rank}(\boldsymbol{Z}) \leq 2,$ and $\boldsymbol{Z} \succeq 0$, which also reaches the minimum of the sum $\sum_{i=3}^{2N} \lambda_i(\boldsymbol{Z})$. Therefore problem (3.7) can be expressed as an optimization problem with the same constraints and an objective function of minimizing the sum $\sum_{i=3}^{2N} \lambda_i(\boldsymbol{Z})$, and the sum $\sum_{i=3}^{2N} \lambda_i(\boldsymbol{Z})$ has an equivalent expression as shown in Theorem 3.3.2.

Theorem 3.3.2. *Given $\lambda_i(\boldsymbol{Z})$ as defined above where $i \in \{1, 2, ..., 2N\}$, we have*

$$\sum_{i=3}^{2N} \lambda_i(\boldsymbol{Z}) = \min_{\boldsymbol{W} \in \Phi} tr(\boldsymbol{W}\boldsymbol{Z}), \tag{3.8}$$

where the set Φ is defined as in [74]:

$$\Phi = \left\{ \boldsymbol{W} \in \mathbb{S}^{2N} : 0 \preceq \boldsymbol{W} \preceq I, tr(\boldsymbol{W}) = 2N - 2 \right\}, \tag{3.9}$$

which is the convex hull of the rank-$(2N-2)$ projection matrices.

Therefore the problem (3.7) can be rewritten as the following when such matrix $\boldsymbol{Z}$ exists:

$$\begin{aligned} \min_{\boldsymbol{Z} \in \mathbb{S}^{2N}} \quad & \min_{\boldsymbol{W} \in \Phi} \text{tr}(\boldsymbol{W}\boldsymbol{Z}) \\ \text{s.t.} \quad & \text{tr}\left(\boldsymbol{M}_j \boldsymbol{X}\right) = z_j, \ \ \forall j \in \mathcal{M}, \\ & \boldsymbol{X} \succeq 0. \end{aligned} \tag{3.10}$$

The condition under which the equivalence is guaranteed, is stated as follows.

Theorem 3.3.3. *The rank of the matrix $\boldsymbol{X}$ is 1, if and only if there exists a $\boldsymbol{W} \in \Phi$, such that*

$$tr(\boldsymbol{W}\boldsymbol{Z}) = 0. \tag{3.11}$$

Recall that the set Φ is defined in (3.9) *in Theorem 3.3.2.*

In problem (3.10), if matrix $\boldsymbol{W}^{opt}$ is fixed, it becomes

$$\begin{aligned} \min_{\boldsymbol{Z} \in \mathbb{S}^{2N}} \quad & \text{tr}(\boldsymbol{W}^{\boldsymbol{opt}}\boldsymbol{Z}) \\ \text{s.t.} \quad & \text{tr}\left(\boldsymbol{M}_j \boldsymbol{X}\right) = z_j, \ \ \forall j \in \mathcal{M}, \\ & \boldsymbol{X} \succeq 0, \end{aligned} \tag{3.12}$$

which is convex optimization problem. Similarly, for any fixed matrix $\boldsymbol{Z}$ obtained by

solving the problem (3.12), denoted as $\boldsymbol{Z}^{opt}$, problem (3.10) becomes

$$\min_{\boldsymbol{W}\in\Phi} \text{tr}(\boldsymbol{W}\boldsymbol{Z}^{opt}), \tag{3.13}$$

which is also convex. Therefore, the solution of the problem (3.10) can be obtained by iteratively solving the two convex problem (3.12) and (3.13).

Assume the solution of the problem (3.7) is $\boldsymbol{X}_{sol}$. There are many methods aiming at recovering the vector $\boldsymbol{v}$. In this chapter, we use the method from [79]:

1). The voltage magnitudes of each bus are determined by $|\boldsymbol{X}_{sol,kk}|$, $k = 1, 2, \ldots, N$;

2). The voltage angles are obtained by the following optimization problem with the assumption of the angle of the reference bus being d_0.

$$\begin{aligned} \min_{\measuredangle\boldsymbol{v}\in[-\pi,\pi]^N} \quad & \sum |\measuredangle\boldsymbol{X}_{st} - (\measuredangle\boldsymbol{v}_s - \measuredangle\boldsymbol{v}_t)| \\ \text{s.t.} \quad & \measuredangle\boldsymbol{v}_{ref} = d_0. \end{aligned}$$

3.4 Power System State Estimation

If errors exist, residual terms will be added to the constraints and we get

$$\text{tr}\,(\boldsymbol{M}_j\boldsymbol{X}) + r_j = z_j, \;\; \forall j \in \mathcal{M},$$

where r_j is the residual for each measurement z_j, and the measurements not only consist of the net active and reactive powers, but also the branch active and reactive powers.

In order to get robust state estimation of the matrix $\boldsymbol{X}$, we consider the following prob-

lem

$$\begin{aligned} \min \quad & \sum_{j=1}^{M} |r_j| \\ \text{s.t.} \quad & \text{tr}\,(\boldsymbol{M}_j \boldsymbol{X}) + r_j = z_j \ \ \forall j \in \mathcal{M}, \\ & \boldsymbol{X} = \boldsymbol{v}\boldsymbol{v}^*. \end{aligned} \tag{4.14}$$

Using the same strategy as in Theorem 3.3.1, we can get the equivalent real version of the problem (4.14). The proof is skipped as it will be the same as in Theorem 3.3.1.

Proposition 3.4.1. *If we use the same notation in* (3.6) *in Proposition 3.3.1, then problem* (4.14) *is equivalent to*

$$\begin{aligned} \min \quad & \sum_{j=1}^{M} |r_j| \\ \textit{s.t.} \quad & \textit{tr}\,(\boldsymbol{M}_j \boldsymbol{X}) + r_j = z_j, \ \ \forall j \in \mathcal{M}, \\ & \textit{rank}(\boldsymbol{Z}) = 2, \\ & \boldsymbol{X} \succeq 0. \end{aligned} \tag{4.15}$$

If we can get the solution of the problem (4.15), by using the same strategy as in Section 3.3, the vector $\boldsymbol{v}$ can be recovered. To solve problem (4.15), similarly as in power flow analysis, we express it as the following equivalent version as long as the condition in Theorem 3.3.3 holds.

$$\begin{aligned} \min_{\boldsymbol{Z} \in \mathbb{S}^{2N}} \quad & \min_{\boldsymbol{W} \in \Phi} \sum_{j=1}^{M} |r_j| + \text{tr}(\boldsymbol{W}\boldsymbol{Z}) \\ \text{s.t.} \quad & \text{tr}\,(\boldsymbol{M}_j \boldsymbol{X}) + r_j = z_j, \ \ \forall j \in \mathcal{M}, \\ & \boldsymbol{X} \succeq 0. \end{aligned} \tag{4.16}$$

For any fixed matrix $\boldsymbol{W}^{opt}$, problem (4.16) becomes

$$\begin{aligned} \min_{\boldsymbol{Z}} \quad & \text{tr}(\boldsymbol{W}^{\boldsymbol{opt}}\boldsymbol{Z}) + \sum_{j=1}^{M} |r_j| \\ \text{s.t.} \quad & \text{tr}\left(\boldsymbol{M}_j\boldsymbol{X}\right) + r_j = z_j, \ \ \forall j \in \mathcal{M}, \\ & \boldsymbol{X} \succeq 0, \end{aligned} \tag{4.17}$$

which is convex. For any fixed matrix $\boldsymbol{Z}$ obtained by solving the problem (4.17), denoted as $\boldsymbol{Z}^{opt}$, problem (4.16) becomes

$$\min_{\boldsymbol{W}\in\Phi} \text{tr}(\boldsymbol{W}\boldsymbol{Z}^{opt}) + \sum_{j=1}^{M} |z_j - \text{tr}\left(\boldsymbol{M}_j\boldsymbol{X}^{opt}\right)|, \tag{4.18}$$

which is also a convex problem. Therefore, the solution of the problem (3.10) can be obtained by iteratively solving the two convex problem (4.17) and (4.18).

3.5 Convergence Analysis

The only difference between the formulation of our method in the power flow analysis and robust state estimation is whether the residual term exists or not, which in this case does not affect the convergency. If the convergence property of our method holds in the power flow analysis so does it in robust state estimation. Therefore we do not distinguish them in this section and will give conclusions in general. But the Appendix C.6 will cover the proofs in the circumstance of state estimation in case of doubts. All proofs are relegated to the appendix.

First of all, we show that the convex iteration can achieve the local optimality as defined in [74].

Theorem 3.5.1. *Assume the initial value of matrix $\boldsymbol{W}$ is $\boldsymbol{W}^{(0)}$. The solutions of $\boldsymbol{W}$ and $\boldsymbol{Z}$ in the k-th iteration are denoted as $\boldsymbol{W}^{(k)}$ and $\boldsymbol{Z}^{(k)}$, respectively. Then, the series $\left\{tr(\boldsymbol{W}^{(k)}\boldsymbol{Z}^{(k)})\right\}$ is bounded and decreasing, thus convergent.*

Remark: State estimation version of Theorem 3.5.1 is shown in the following Thoerem 3.5.2, and proof can be found in the Appendix.

Theorem 3.5.2. *Assume the initial value of matrix $\boldsymbol{W}$ is $\boldsymbol{W}^{(0)}$. The solutions of $\boldsymbol{W}$ and $\boldsymbol{Z}$ in the k-th iteration are denoted as $\boldsymbol{W}^{(k)}$ and $\boldsymbol{Z}^{(k)}$, respectively. Then, the series $\left\{ \sum_{j=1}^{M} |r_j^{(k)}| + tr(\boldsymbol{W}^{(k)}\boldsymbol{Z}^{(k)}) \right\}$ is bounded and decreasing, thus convergent.*

Next we show that such matrix $\boldsymbol{W}$ in Theorem 3.3.3 exists.

Theorem 3.5.3. *Suppose $\boldsymbol{Z}$ is the solution of the problem* (3.7)*. Then there exists a matrix $\boldsymbol{W}$ satisfying condition* (3.11) *and* (3.9)*.*

One can always choose the zero matrix $\mathbf{0}$ or identity matrix $\boldsymbol{I}$ as the initial value of the matrix $\boldsymbol{W}$ if there is no better way to initialize it. Assume the global optimal value of the problem is $\boldsymbol{Z}_{opt}$. The *optimal projection direction*, denoted as $\boldsymbol{W}_{opd}$ is defined as the matrix $\boldsymbol{W} \in \Phi$ such that $\text{tr}(\boldsymbol{W}\boldsymbol{Z}_{opt}) = 0$. If $\boldsymbol{W}_{opd}$ is a solution of the convex minimization problem in the iterations, then according to Theorem 3.5.1, the local optimality would reach the global optimal value, 0. In this case, the local optimum is actually the global optimum.

3.6 Numerical Tests

This section presents the numerical results for robust state estimation in addition to power flow analysis. The datasets used here are all from Matpower [76]. Two methods, the method in Zhang et al. [67] and the weighted least squares via Newton's method, are used for comparison to verify the advantage of our method. For simplification, the weighted least squares via Newton's method is denoted as WLS. The performance of each method is measured by the root-mean-square error (RMSE) of the estimated voltage $\hat{\boldsymbol{v}}$, defined as

$$\text{RMSE}(\hat{\boldsymbol{v}}) = \frac{1}{\sqrt{N}}\|\boldsymbol{v} - \hat{\boldsymbol{v}}\|,$$

where $\boldsymbol{v}$ is the true voltage, and N is the length of the vector $\boldsymbol{v}$, or the number of buses in the power network in our case. For the simplification of all calculations, we always use the

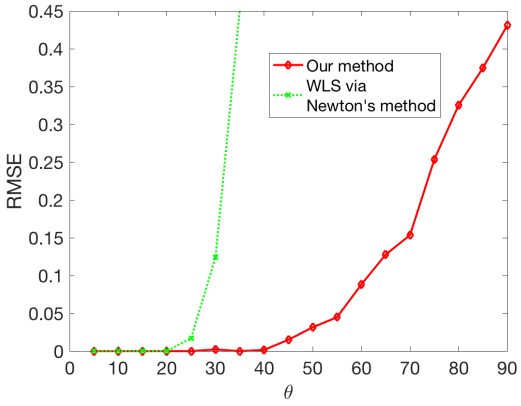
flat start, the identity matrix as the initial start value of the matrix $\boldsymbol{X}$ in our algorithm. In Zhang's method, they provide several choices for designing the matrix $\boldsymbol{M_0}$: $\boldsymbol{Y^*Y}$, $-\boldsymbol{B}$, $\alpha\boldsymbol{I} - \boldsymbol{B}$, real symmetric matrix with negative values at entries corresponding to the line flow measurements and zero elsewhere, and entries corresponding to the line flow (s,t) measurements being $-\boldsymbol{B}_{st}$ and $\boldsymbol{M}_{0;ii} = \sum_{j=1}^{N} |\boldsymbol{B}_{ij}|$ for all $i \in \{1,\ldots,N\}$. We run them all and select the one that has the best result.

The rest of this section is organized as follows. Subsection 3.6.2 shows simulation results for state estimation problems. We conduct noise study, bad data study,and start value study in this subsection to compare our method with Zhang's method and WLS method in different scenarios. In subsection 3.6.1, we show that our method can also be applied to power flow analysis.

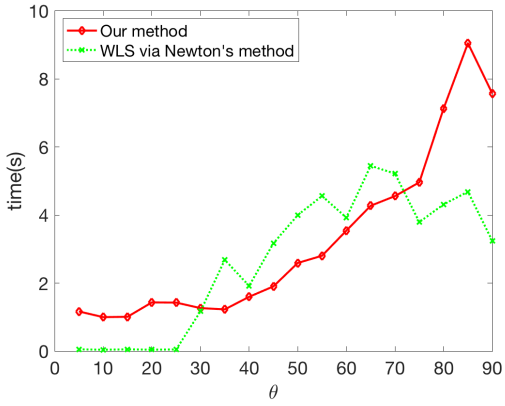
3.6.1 Power Flow (PF) Simulation Results

The known measurements for the classical power flow problem (PF) are 1). Voltage magnitudes for the reference bus and the PV buses; 2). Real power measurements for PV buses; 3). Reactive power measurements for PQ buses. The voltage phase for the reference bus is also given. The true voltage magnitudes are chosen from the uniform distribution Unif$[0.9, 1.1]$, and the true voltage angles (degrees) are from uniform distribution Unif$[-\theta, \theta]$, where $\theta = 5, 10, 15, \ldots, 90$. For each θ, we repeat the simulation for 50 times, and record the average values of RMSE and time for our method and the classical WLS method. Figure 3.1, 3.2, and 3.3 show the simulation results for the 9-, 30-, and 57- bus systems.

Overall, WLS performs faster than our method, however, when θ goes beyond 5 or 10 degrees, it diverges quickly and fails to find the true solutions. Although our algorithm consumes more time, it is capable of keeping the accuracy when θ is moderately large.

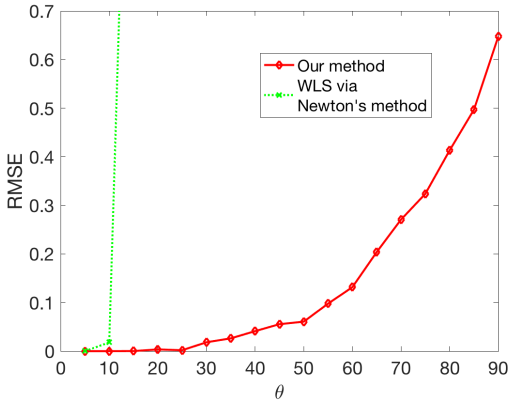


Mean RMSE

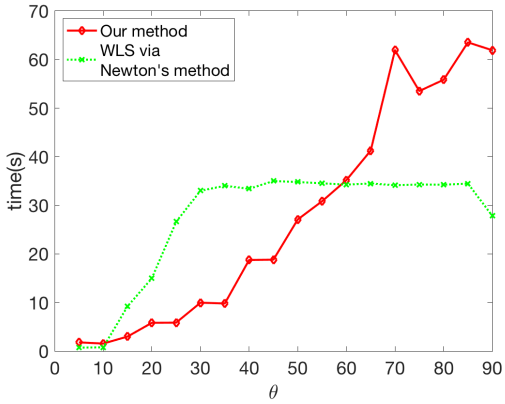


Mean Time

Figure 3.1: 9-bus system power flow problem simulation

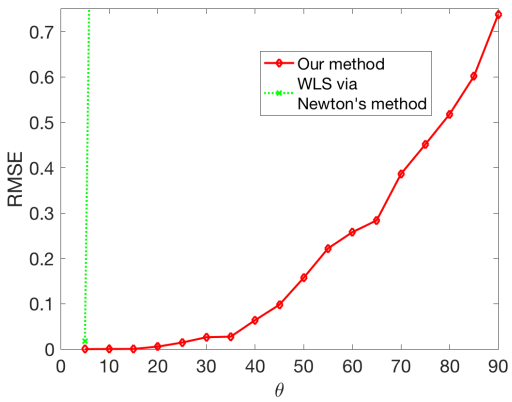


Mean RMSE

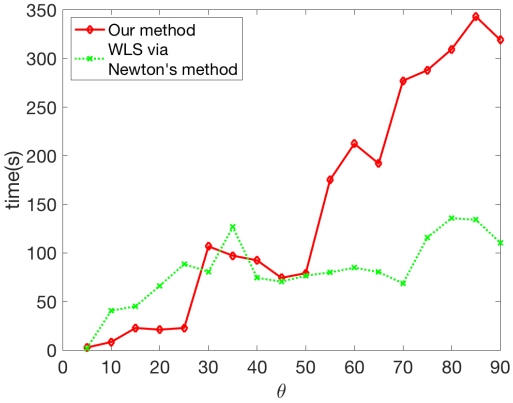


Mean Time

Figure 3.2: 30-bus system power flow problem simulation



Mean RMSE



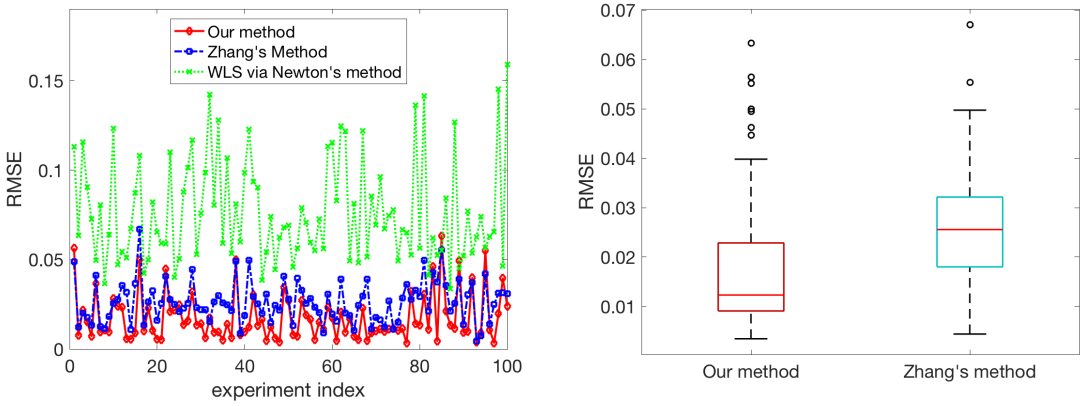
Mean Time

Figure 3.3: 57-bus system power flow problem simulation

3.6.2 Power System State Estimation (PSSE) Simulation Results

Comparison with Zhang's Method

First of all, we conduct our method under the same circumstances as in [67]. The measurements are 1). voltage magnitudes for all buses; 2). active power flows at both ends of all lines of a spanning tree of the network. Zero-mean Gaussian noises are added with 0.002 and 0.001 per unit standard deviations for squared voltage magnitudes and line flows respectively. Besides, 20% of randomly chosen line flow measurements are generated as bad data, which are contaminated by adding zero-mean Gaussian noises with 0.1 per unit standard deviation. We repeat the simulation for 100 times. Figure 3.4 and 3.5 show the simulation results for the 57- and 118- bus systems. Both our method and Zhang's method outperform WLS in these situations. Our method is able to achieve smaller RMSE with smaller variation.

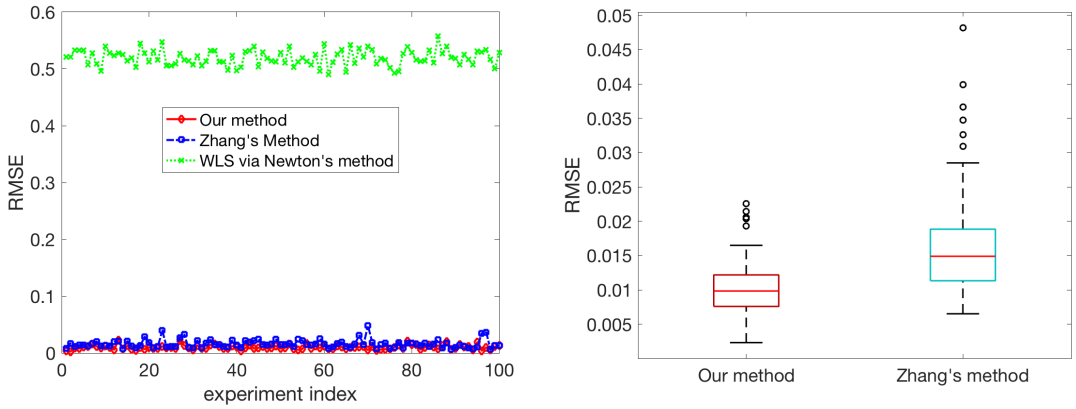


RMSE Comparison Boxplot

Figure 3.4: Our method vs. Zhang's method vs. WLS in the 57-bus system.

Noise Study

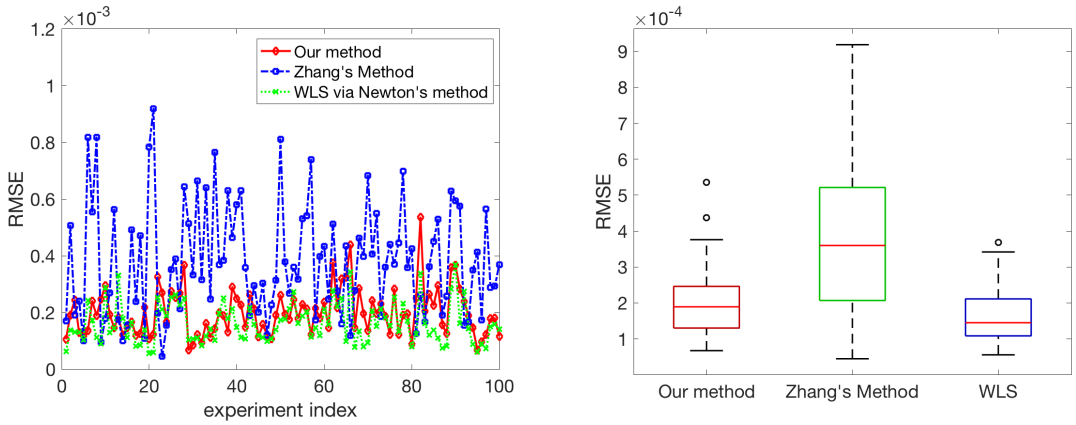
In this part, we introduce Gaussian noise terms for all of the measurements for the 9-, 57-, and 118- bus systems. Particularly for the 9-bus system, we introduce Gaussian noise $N(0, 0.001^2)$ and use the following: 1). Voltage magnitude for the reference bus and the



RMSE Comparison Boxplot

Figure 3.5: Our method vs. Zhang's method vs. WLS in the 118-bus system.

PV buses; 2). Real power constraints for all buses; 3). Reactive power constraints for the PQ buses; 4). From-end branch real and reactive power constraints for branch 1,3,4,5,7; 5). To-end branch real and reactive power constraints for branch 7,8. We conduct the simulation for 100 times. Figure 3.6 shows the simulation results for it. Zhang's method is worse than the classical WLS and our method, while ours can get the comparably good solutions as WLS.

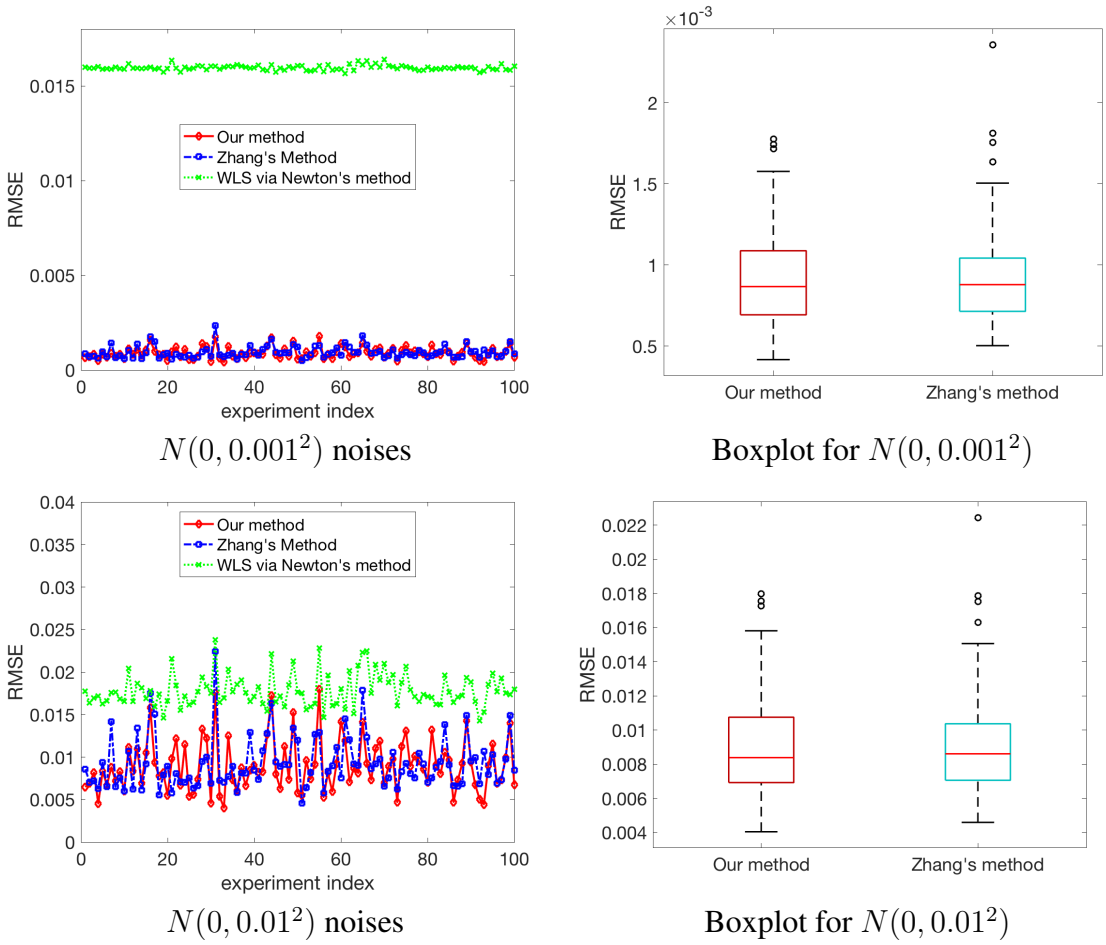


$N(0, 0.001^2)$ noises Boxplot

Figure 3.6: Simulation results with noises in the 9-bus system.

For the 57-bus systems, as in [67] their method are more suitable when all the voltage magnitudes are given, so we select the following measurements to run the simulation in order to make fair comparison. 1). voltage magnitudes at all buses; 2). power flow

measurements per line of a minimal spanning tree. We introduce the $N(0, 0.001^2)$ and the $N(0, 0.01^2)$ noises to all the measurement respectively, and repeat the simulation for 100 times. Figure 3.7 shows the results with different noise variances. In both settings, WLS is worse than the other two, and ours still performs comparably well with Zhang's method, and with slightly better RMSE in terms of both mean and variance.



$N(0, 0.001^2)$ noises

Boxplot for $N(0, 0.001^2)$

$N(0, 0.01^2)$ noises

Boxplot for $N(0, 0.01^2)$

Figure 3.7: Simulation results with $N(0, 0.001^2)$ and $N(0, 0.01^2)$ noises in the 57-bus system.

For the 118-bus system, we go back to the general setting as follows: 1). Voltage magnitude for the reference bus and the PV buses; 2). Real power constraints for all buses; 3). Reactive power constraints for the PQ buses; 4). From-end branch real and reactive power constraints; 5). To-end branch real and reactive power constraints. The simulation results are shown in Figure 3.8. Without the specific voltage magnitude constraints for

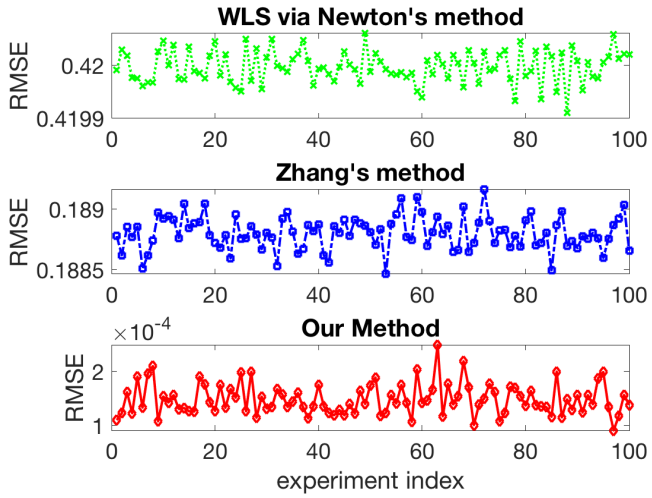


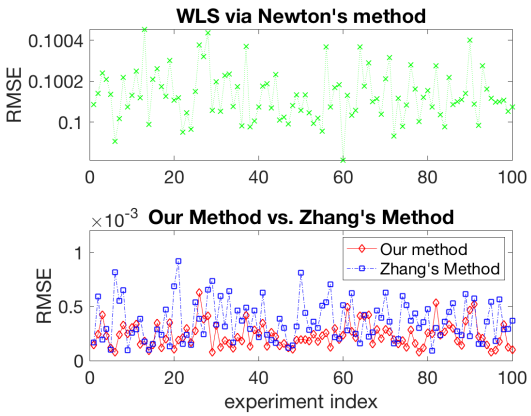
Figure 3.8: Simulation results with $N(0, 0.001^2)$ noises in the 118-bus system.

all buses, Zhang's method does not perform well although still better than WLS, while our method outperforms the other two and remains the same order of RMSE as previous situations.

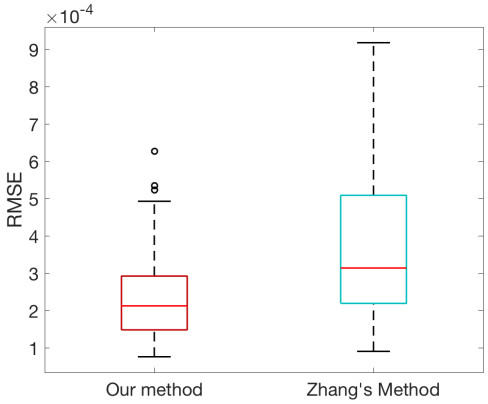
Bad Data Study

Following the same structure as in the previous noise study section 3.6.2, in additional to noises, we introduce bad data to the measurement as well. The selected measurements for each bus system are the same as previous study. In the 9-bus system, we conduct two different bad data situations: (i). an error of $N(-2, 0.001^2)$ is added to a randomly selected measurement (the reactive power of the to-end of the branch 8), in addition to the $N(0, 0.001^2)$ noises for all measurements; (ii) in addition to the error introduced in (i), we add one more error of $N(2, 0.001^2)$ to another randomly selected measurement (the real power of the from-end of the branch 5). Figure 3.9 shows the simulation results for both situations. WLS does not perform well when bad data exist, while both our method and Zhang's method can successfully recover a good state estimation, and ours has a better estimate with smaller RMSE and less variance.

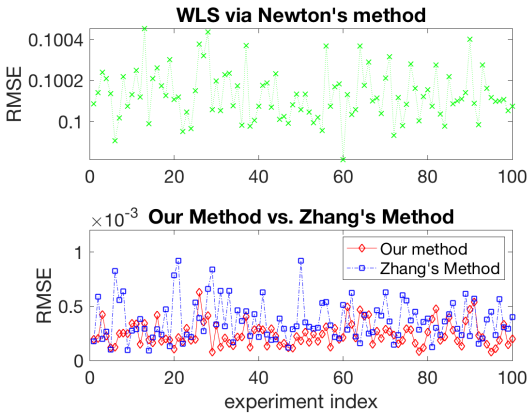
In the 57-bus system, apart from the $N(0, 0.001^2)$ noises, we add ten randomly selected



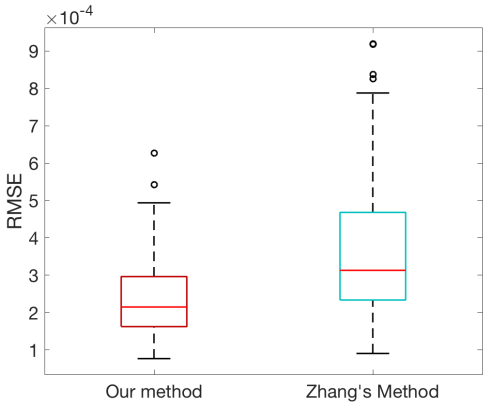
One Error $N(-2, 0.001^2)$



Our Method vs. Zhang's Method



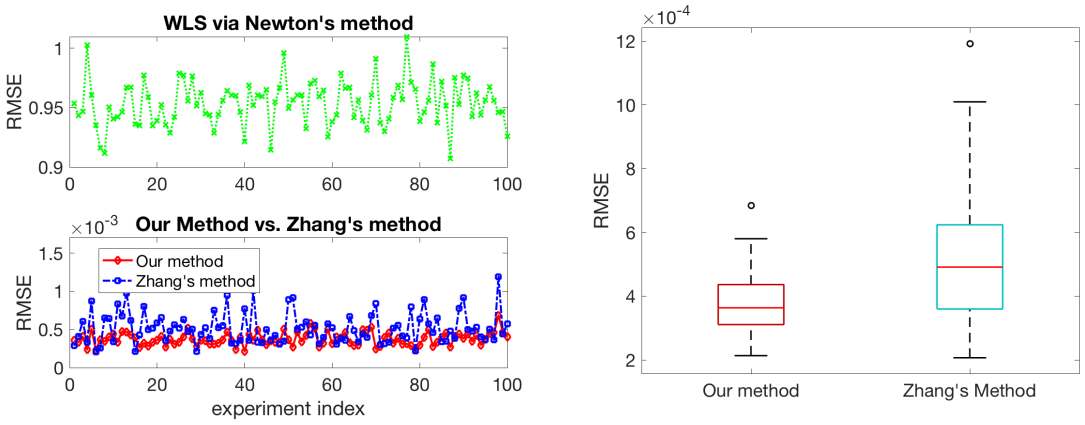
Two Errors $N(\mp 2, 0.001^2)$



Our Method vs. Zhang's Method

Figure 3.9: Bad data simulation results with $N(0, 0.001^2)$ noises in the 9-bus system.

bad data to the measurements. The bad data follows Gaussian distribution $N(2, 0.001^2)$. Figure 3.10 is the simulation summary. Both our method and Zhang's method are significantly better than WLS, while our method shows slightly better properties on the solutions.



Ten Errors $N(2, 0.001^2)$ Our Method vs. Zhang's Method

Figure 3.10: Bad data simulation results with $N(0, 0.001^2)$ noises in the 57-bus system.

For the 118-bus system, similarly, we also add ten randomly selected bad data to the measurements with Gaussian distribution $N(2, 0.001^2)$. Figure 3.11 gives the simulation results for the three methods with similar pattern as in Figure 3.8. Our method shows advantages over WLS and Zhang's method.

Start Value Study

In simulations, it is critical to find the right $\boldsymbol{M}_0$ matrix as the start value in Zhang's method in order to get correct solutions. The searching process needs a lot of efforts and in reality will bring risks and uncertainty to the solutions if the right matrix $\boldsymbol{M}_0$ is not found. However, the flat start, identity matrix works in all the simulation we run. Figure 3.12 gives the demonstration of influences of different start values on the results of our method based on the 57-bus system with $N(0, 0.001^2)$ noises added to all constraints. In this case, we randomly pick the values between 0 and 1 for each entry of the starting matrix $\boldsymbol{W}$, and repeat for 100 times. Simulation results show that our method is not sensitive to the starting value compared to Zhang's method. Other cases also show the similar patterns.

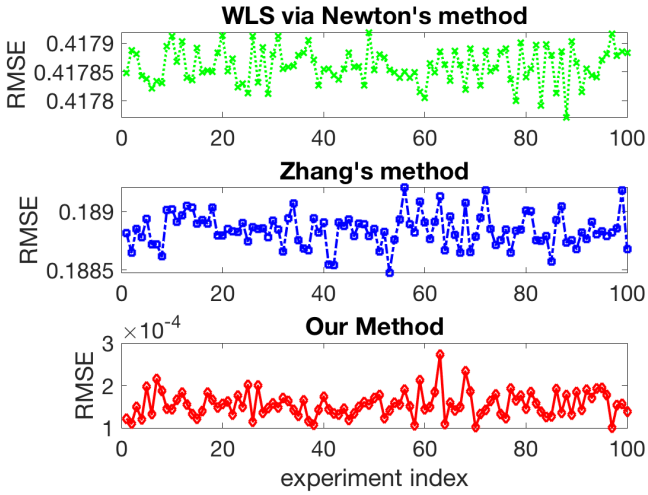


Figure 3.11: Ten errors with $N(0, 0.001^2)$ noises in the 118-bus system.

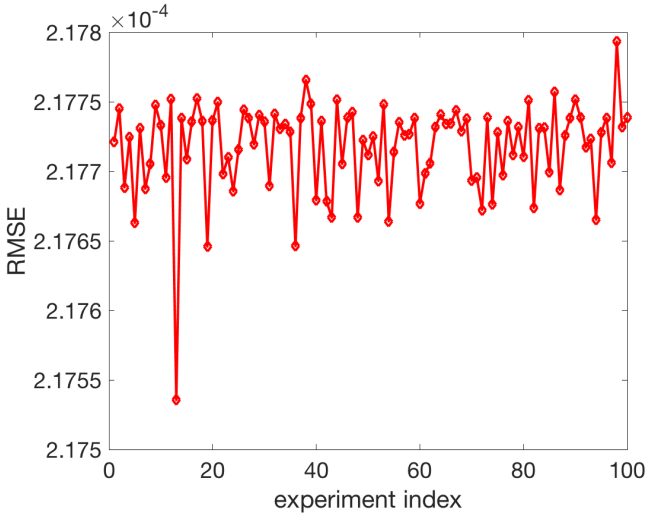


Figure 3.12: Different start values with $N(0, 0.001^2)$ noises in the 57-bus system.

3.7 Conclusion

In this chapter, we develop a new semidefinite programming algorithm for both the power flow and power system state estimation problems. We first formulates the two complex-valued problems into two non-convex problems with real-valued objective functions and semidefinite constraints. Rather than doing convex relaxation to overcome the disadvantages of nonconvexity, we instead formulate it as a sequence optimization problem, which is to solve two well-defined convex problems and avoids the matrix preconstruction in the objective function of the convex relaxation programming such as in the method in [67]. This makes our algorithm more adaptable and applicable in complicated or uninformative circumstances. In order to show the feasibility and convergency, we provide convergence analysis for our new algorithm and the condition when equivalency holds between the sequence optimization problem and the non-convex semidefinite programming problem. Furthermore, by conducting simulations on the classical power flow systems, iteratively method shows comparable performance in the situations that other methods - the convex relaxation method in [67], and the weighted least squares method via Newton's method - are applicable, such as situations when all voltage magnitudes are known, situations when the voltage angles are small. Moreover, in other situations, such as not all voltage magnitudes are known or the voltage angles are not close to zero, our method still works and outperforms the others. Furthermore, the simulation results show that our method has good performance when bad data exist. Overall, we are confident to recommend our newly proposed method in both power flow analysis and power system state estimation, especially when the voltage angles could be large or bad data and/or noises exist.

Appendices

APPENDIX A

PROOFS IN CHAPTER 1

A.1 Proof of Lemma 1.2.8

Proof. As W is a subspace of $\mathbb{R}^p$ satisfying $W^\perp \subset W_X^\perp$, we have

$$W \supset W_X, \text{ and } W \setminus W_X \subset W_X^\perp.$$

Because $P_{W_X}^\perp X \perp\!\!\!\perp Y$ according to our Assumption 1.2.6, we know

$$P_{W \setminus W_X} X \perp\!\!\!\perp Y.$$

Therefore, based on (2.4), we know that the minimal value of $\mathcal{V}^2(u^T X', Y)$ is 0, which indicates $P_{u^*} X \perp\!\!\!\perp Y$. Therefore u^* can only come from the subspace $W \setminus W_X \subset W_X^\perp$.

Apparently $u^* \notin W^\perp$ since we ahve $u^* \in W$ as shown above, which implies that u^* is not any linear combination of $S^\perp$. So the dimension of the subspace of $K^\perp = \text{span}\left(\{Su^*\} \cup S^\perp\right)$ is larger than the dimension of the subspace $W^\perp$, which can also be written as

$$W^\perp \subset K^\perp \subseteq W_X^\perp.$$

□

A.2 Proof of Lemma 1.2.11

Proof. If we define function $f(u; \mathbf{X}, \mathbf{Y})$ as

$$f(u; \mathbf{X}, \mathbf{Y}) = \sum_{i,j=1}^{N} g_{ij} |u^T (X_i - X_j)|,$$

where g_{ij} are defined in the Lemma 1.2.11. Based on the equation (2.18) in [20], we can rewrite $\mathcal{V}_N^2(\mathbf{X}u, \mathbf{Y})$ as

$$\mathcal{V}_N^2(\mathbf{X}u, \mathbf{Y}) = \frac{1}{N^2} f(u; \mathbf{X}, \mathbf{Y}).$$

The detailed calculation is stated as follows.

$$\begin{aligned}
\mathcal{V}_N^2(\mathbf{X}u, \mathbf{Y}) =& \frac{1}{N^2}\sum_{i,j=1}^{N} |u^T(X_i - X_j)||Y_i - Y_j|_q - \frac{2}{N^3}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{m=1}^{N} |u^T(X_i - X_j)||Y_i - Y_m|_q \\
&+ \frac{1}{N^2}\sum_{i,j=1}^{N} |u^T(X_i - X_j)| \frac{1}{N^2}\sum_{i,j=1}^{N} |Y_i - Y_j|_q \\
=& \frac{1}{N^2}\sum_{i,j=1}^{N} |u^T(X_i - X_j)| \left(|Y_i - Y_j|_q - \frac{2}{N}\sum_{m=1}^{N} |Y_i - Y_m|_q + \frac{1}{N^2}\sum_{i,j=1}^{N} |Y_i - Y_j|_q \right) \\
=& \frac{1}{N^2}\sum_{i,j=1}^{N} |u^T(X_i - X_j)| \Bigg(|Y_i - Y_j|_q - \frac{1}{N}\sum_{k=1}^{N} |Y_i - Y_k|_q - \frac{1}{N}\sum_{k=1}^{N} |Y_j - Y_k|_q \\
&+ \frac{1}{N^2}\sum_{k,l=1}^{N} |Y_k - Y_l|_q \Bigg) \\
=& \frac{1}{N^2}\sum_{i,j=1}^{N} g_{ij} |u^T(X_i - X_j)| \\
=& \frac{1}{N^2} f(u; \mathbf{X}, \mathbf{Y}).
\end{aligned}$$

From another point of view, as in Definition 1.2.1, for any two random vectors $X \in \mathbb{R}^p$, $Y \in \mathbb{R}^q$, and a pre-specified direction $u \in \mathbb{R}^p$,

$$\begin{aligned}
\mathcal{V}^2(u^T X, Y) \;=\; & \mathbb{E}\left[|u^T(X - X')||Y - Y'|_q\right] - \mathbb{E}\left[|u^T(X - X')||Y - Y''|_q\right] \\
& - \mathbb{E}\left[|u^T(X - X'')||Y - Y'|_q\right] + \mathbb{E}\left[|u^T(X - X')|\right] \mathbb{E}\left[|Y - Y'|_q\right].
\end{aligned}$$

Define

$$g(X, X') = \mathbb{E}\left[|Y - Y'|_q - |Y - Y''|_q - |Y' - Y''|_q + \mathbb{E}\left[|Y - Y'|_q\right] \middle| X, X'\right]. \quad (2.1)$$

Then the function $\mathcal{V}^2(u^T X, Y)$ can be rewritten as

$$\mathcal{V}^2(u^T X, Y) = \mathbb{E}\left[g(X, X')|u^T(X - X')|\right]. \tag{2.2}$$

Now we consider the sample version. From (2.1), an estimate of $g(X, X')$ can be, for given $X = X_i$, and $X' = X_j$,

$$g_{ij} = |Y_i - Y_j|_q - \frac{1}{N}\sum_{k=1}^{N}|Y_i - Y_k|_q - \frac{1}{N}\sum_{k=1}^{N}|Y_j - Y_k|_q + \frac{1}{N^2}\sum_{k,l=1}^{N}|Y_k - Y_l|_q,$$

which further gives us an estimate of $\mathcal{V}^2(u^T X, Y)$.

Note that all g_{ij}'s $(i, j = 1, \cdots, N)$ can be computed and one can verify the following properties of g_{ij}'s:

1. for any $1 \leq i \neq j \leq N$, we have $g_{ij} = g_{ji}$, i.e., g_{ij}'s are symmetric subject to the subscripts switching;
2. the total sum of g_{ij}'s is equal to zero: $\sum_{i,j=1}^{N} g_{ij} = 0$.

Applying the definition of M_+ and M_- to the function $f(u; \mathbf{X}, \mathbf{Y})$, we have

$$f(u; \mathbf{X}, \mathbf{Y}) = \sum_{i,j=1}^{N} g_{ij}|u^T(X_i - X_j)| = 2\sum_{i,j=1, j>i}^{N} g_{ij}|u^T(X_i - X_j)| = 2\left(\|M_+ u\|_1 - \|M_- u\|_1\right)$$

Omitting the constant $\frac{1}{N^2}$ and 2, we have the equivalent version of Problem 2.6:

$$\begin{aligned} \min_{u \in \mathbb{R}^p} \quad & \|M_+ u\|_1 - \|M_- u\|_1 \\ \text{subject to:} \quad & \|u\|_2 = 1. \end{aligned}$$

□

A.3 Proof of Lemma 1.3.1

Proof. Suppose the statement, $u \to u^*$, as $N \to \infty$ is not true. Then we can select a subsequence $\{u_N\}$ such that $\lim_{N\to\infty} u_N = u'$, where $u' \neq \text{argmin}\left\{\mathcal{V}^2(u^T X', Y) : u \in W', \|u\|_2 = 1\right\}$. Therefore,

$$\mathcal{V}^2((u')^T X', Y) > \mathcal{V}^2((u^*)^T X', Y). \tag{3.3}$$

As $u_N = \text{argmin}\left\{\|M_+ u\|_1 - \|M_- u\|_1 : u \in W', \|u\|_2 = 1\right\}$, we have

$$\frac{1}{N^2}\left(\|M_+ u_N\|_1 - \|M_- u_N\|_1\right) < \frac{1}{N^2}\left(\|M_+ u^*\|_1 - \|M_- u^*\|_1\right). \tag{3.4}$$

Since $\frac{1}{N^2}\left(\|M_+ u_N\|_1 - \|M_- u_N\|_1\right)$ is a continuous function, we have

$$\lim_{u_N \to u'} \frac{1}{N^2}\left(\|M_+ u_N\|_1 - \|M_- u_N\|_1\right) = \frac{1}{N^2}\left(\|M_+ u'\|_1 - \|M_- u'\|_1\right). \tag{3.5}$$

According to Theorem 1.2.4 and (3.5), letting $N \to \infty$ on both sides of equation (3.4), we can get

$$\mathcal{V}^2((u')^T X', Y) < \mathcal{V}^2((u^*)^T X', Y).$$

This is a contradiction to (3.3), which implies our assumption is not true. Therefore, we have $u \to u^*$, as $N \to \infty$. □

A.4 Proof of Theorem 1.3.3

Proof. Let $\widehat{U}^\perp$ be the orthonormal basis of $\widehat{W}_X^\perp$ – the orthogonal complement of $\widehat{W}_X$. From Lemma 1.3.1 we know that each $\hat{u}_i$ in $\widehat{U}^\perp$ is convergent to some unit vector, u_i in $W_X^\perp$, where $U^\perp = [u_1, u_2, \dots]$ is the orthonormal basis of $W_X^\perp$.

There exist matrix $\widehat{U}$ and U, such that $[\widehat{U}^\perp, \widehat{U}]$, $[U^\perp, U]$ are p-by-p orthonormal matrix,

where span(U) is an orthonormal basis of W_X. Then,

$$\text{dist}(\widehat{W}_X, W_X) = \| \left(\widehat{U}^\perp\right)^T U\|_2.$$

As $\widehat{U}^\perp \to U^\perp$ as $N \to \infty$, we have

$$\lim_{N\to\infty} \text{dist}(\widehat{W}_X, W_X) = \| \left(U^\perp\right)^T U\|_2 = 0.$$

□

A.5 Proof of Theorem 1.3.4

Proof. Theorem 6 in [20] shows that

for all $0 < \alpha < 0.215$,

$$\lim_{N\to\infty} 1 - P_N^{(t)} \leq \alpha, \text{ and } \sup_{u^T X \perp\!\!\!\perp Y} \left\{ \lim_{N\to\infty} 1 - P_N^{(t)} \right\} = \alpha.$$

Let $\gamma = 1 - \alpha$. The theorem is proved. □

A.6 Proof of Lemma 1.4.2

Proof. The proof is based on Prof. Udell's ORIE 6326 slides at Cornell and Prof. Lieven Vandenberghe's EE236 slides. The right-hand side can be equivalently written as $f^*(y) + f(x) = y^T x$ by the definition of conjugate function. Therefore, the proof is finished if the following holds

$$x \in \partial f^*(y) \iff f^*(y) + f(x) = y^T x. \tag{6.6}$$

When $f : \mathbb{R}^p \to \mathbb{R}$ is lower semi-continuous and convex, the following two propositions holds, which leads to Equation (6.6).

Proposition A.6.1. $y \in \partial f(x) \iff f^*(y) + f(x) = y^T x$.

Proposition A.6.2. $y \in \partial f(x) \Longleftrightarrow x \in \partial f^*(y)$.

□

A.7 Proof of Proposition B.3.2

Proof. By definition of subdifferential we have

$$y \in \partial f(x) \Longleftrightarrow y^T x - f(x) \geq y^T z - f(z), \forall z \Longleftrightarrow y^T x - f(x) \geq \sup_z \{y^T z - f(z)\} = f^*(y).$$

By the definition of conjugate function, we have $f^*(y) \geq y^T x - f(x)$. Therefore the following is proved:

$$y \in \partial f(x) \Longleftrightarrow f^*(y) + f(x) = y^T x.$$

□

A.8 Proof of Proposition B.3.1

Proof. If $y \in \partial f(x)$, according to Proposition B.3.2, we can get $f^*(y) = y^T x - f(x)$. Therefore, for any z, we have

$$f^*(z) = \sup_t \{z^T t - f(t)\} \geq z^T x - f(x) = x^T(z-y) + y^T x - f(x) = x^T(z-y) + f^*(y),$$

which indicates $x \in \partial f^*(y)$. As $f^{**} = f$, we can get $y \in \partial f(x)$ if $x \in \partial f^*(y)$. □

A.9 Proof of Lemma 1.4.3

Proof. From (4.12), we know

$$\begin{aligned}
u_{l+1} &= \text{argmin} L_\rho(u, z_l, v_l) \\
&= \text{argmin} \left\{ \frac{\xi^{(t)}}{2} u^T u - y_k^T u + \|z_l\|_1 + v_l^T (M_+ u - z_l) + \frac{\rho}{2} \|M_+ u - z_l\|_2^2 \right\} \\
&= \text{argmin} \left\{ \frac{\xi^{(t)}}{2} u^T u - y_k^T u + v_l^T M_+ u + \frac{\rho}{2} \|M_+ u - z_l\|_2^2 \right\} \\
&= \text{argmin} \left\{ \frac{\xi^{(t)}}{2} u^T u - y_k^T u + v_l^T M_+ u + \frac{\rho}{2} \left(u^T M_+^T M_+ u - 2 z_l^T M_+ u \right) \right\} \\
&= \text{argmin} \left\{ u^T \left(\frac{\xi}{2} I_p + \frac{\rho}{2} M_+^T M_+ \right) u + \left(M_+^T v_l - y_k - \rho M_+^T z_l \right)^T u \right\}.
\end{aligned}$$

This is a quadratic programming problem, so the minimization is achieved when the following condition is satisfied:

$$0 = 2 \left(\frac{\xi^{(t)}}{2} I_p + \frac{\rho}{2} M_+^T M_+ \right) u_{l+1} + \left(M_+^T v_l - y_k - \rho M_+^T z_l \right).$$

By solving the above equation, we can get

$$u_{l+1} = \left(\xi^{(t)} I_p + \rho M_+^T M_+ \right)^{-1} \left(y_k + M_+^T (\rho z_l - v_l) \right).$$

Again according to (4.12), we know

$$\begin{aligned}
z_{l+1} &= \text{argmin} L_\rho(u_{l+1}, z, v_l) \\
&= \text{argmin} \left\{ \frac{\xi^{(t)}}{2} u_{l+1}^T u_{l+1} + \|z\|_1 - y_k^T u_{l+1} + v_l^T (N u_{l+1} - z) + \frac{\rho}{2} \|M_+ u_{l+1} - z\|_2^2 \right\} \\
&= \text{argmin} \left\{ \frac{\rho}{2} z^T z - (v_l + \rho M_+ u_{l+1})^T z + \|z\|_1 \right\}.
\end{aligned}$$

This is also a convex problem, so it is minimized when the following condition is achieved:

$$0 = 2\frac{\rho}{2}z_{l+1} - (v_l + \rho M_+ u_{l+1}) + \partial\|z_{l+1}\|_1 = \rho z_{l+1} + \partial\|z_{l+1}\|_1 - (v_l + \rho M_+ u_{l+1}),$$

which leads to

$$z_{l+1} = S(\frac{1}{\rho}v_l + M_+ u_{l+1}, \frac{1}{\rho}).$$

□

A.10 Proof of Lemma 1.4.4

Proof. By definition of function $L(u_k; \xi^{(t)}, \psi^{(t)})$, we have

$$\begin{aligned}
& L(u_k; \xi^{(t)}, \psi^{(t)}) - L(u_{k+1}; \xi^{(t)}, \psi^{(t)}) \\
= \; & \frac{\xi^{(t)}}{2}(u_k^T u_k - u_{k+1}^T u_{k+1}) + \left(\xi^{(t)} - \psi^{(t)}\right)\left(\|u_{k+1}\|_2 - \|u_k\|_2\right) \\
& + \|M_+ u_k\|_1 - \|M_+ u_{k+1}\|_1 + \|M_- u_{k+1}\|_1 - \|M_- u_k\|_1 \\
= \; & \frac{\xi^{(t)}}{2}\|u_{k+1} - u_k\|_2^2 + \xi^{(t)}\langle u_k - u_{k+1}, u_{k+1}\rangle + \left(\xi^{(t)} - \psi^{(t)}\right)\left(\|u_{k+1}\|_2 - \|u_k\|_2\right) \\
& + \|M_+ u_k\|_1 - \|M_+ u_{k+1}\|_1 + \|M_- u_{k+1}\|_1 - \|M_- u_k\|_1.
\end{aligned}$$

Because u_{k+1} is the solution of $\min\{\frac{\xi^{(t)}}{2}u^T u + \|M_+ u\|_1 - y_k^T u\}$, we have

$$\xi^{(t)} u_{k+1} + M_+^T \partial\|M_+ u_{k+1}\|_1 - y_k = 0.$$

Multiplied by $(u_k - u_{k+1})^T$, we get

$$\xi^{(t)}\langle u_k - u_{k+1}, u_{k+1}\rangle + \langle u_k - u_{k+1}, M_+^T \partial\|M_+ u_{k+1}\|_1\rangle - \langle u_k - u_{k+1}, y_k\rangle \ni 0.$$

Then we have

$$\xi^{(t)}\langle u_k - u_{k+1}, u_{k+1}\rangle = -\left(\partial\|M_+ u_{k+1}\|_1\right)^T M_+ u_k + \|M_+ u_{k+1}\|_1 + \langle u_k - u_{k+1}, y_k\rangle.$$

Therefore $L(u_k;\xi^{(t)}) - L(u_{k+1};\xi^{(t)})$ can be written as

$$
\begin{aligned}
& L(u_k;\xi^{(t)},\psi^{(t)}) - L(u_{k+1};\xi^{(t)},\psi^{(t)}) \\
= \;& \frac{\xi^{(t)}}{2}\|u_{k+1}-u_k\|_2^2 - \left(\partial\|M_+u_{k+1}\|_1\right)^T M_+u_k + \|M_+u_{k+1}\|_1 + \langle u_k - u_{k+1}, y_k\rangle \\
& + \left(\xi^{(t)}-\psi^{(t)}\right)\left(\|u_{k+1}\|_2 - \|u_k\|_2\right) + \|M_+u_k\|_1 - \|M_+u_{k+1}\|_1 + \|M_-u_{k+1}\|_1 - \|M_-u_k\|_1 \\
= \;& \frac{\xi^{(t)}}{2}\|u_{k+1}-u_k\|_2^2 + \left(\|M_+u_k\|_1 - \left(\partial\|M_+u_{k+1}\|_1\right)^T M_+u_k\right) \\
& + \left(\|M_-u_{k+1}\|_1 + \left(\xi^{(t)}-\psi^{(t)}\right)\|u_{k+1}\|_2\right) - \left(\|M_-u_k\|_1 + \left(\xi^{(t)}-\psi^{(t)}\right)\|u_k\|_2\right) \\
& - \langle u_{k+1}-u_k, y_k\rangle.
\end{aligned}
$$

Since $\|M_+u_k\|_1 \geq \left(\partial\|M_+u_{k+1}\|_1\right)^T M_+u_k$, we have

$$
\begin{aligned}
L(u_k) - L(u_{k+1}) \;\geq\; & \frac{\xi^{(t)}}{2}\|u_{k+1}-u_k\|_2^2 + \left(\|M_-u_{k+1}\|_1 + \left(\xi^{(t)}-\psi^{(t)}\right)\|u_{k+1}\|_2\right) \\
& - \left(\|M_-u_k\|_1 + \left(\xi^{(t)}-\psi^{(t)}\right)\|u_k\|_2\right) - \langle u_{k+1}-u_k, y_k\rangle.
\end{aligned}
$$

Since $y_k \in \partial h(u_k;\xi^{(t)},\psi^{(t)})$, we have

$$
h(u_{k+1};\xi^{(t)},\psi^{(t)}) - h(u_k;\xi^{(t)},\psi^{(t)}) \geq y_k^T(u_{k+1}-u_k),
$$

which is equivalent to

$$
\left(\|M_-u_{k+1}\|_1 + \left(\xi^{(t)}-\psi^{(t)}\right)\|u_{k+1}\|_2\right) - \left(\|M_-u_k\|_1 + \left(\xi^{(t)}-\psi^{(t)}\right)\|u_k\|_2\right) - \langle u_{k+1}-u_k, y_k\rangle \geq 0.
$$

Therefore we get

$$
L(u_k;\xi^{(t)},\psi^{(t)}) - L(u_{k+1};\xi^{(t)},\psi^{(t)}) \geq \frac{\xi^{(t)}}{2}\|u_{k+1}-u_k\|_2^2 \geq 0.
$$

□

A.11 Proof of Theorem 1.4.5

Proof. 1. Since we have

$$L(u;\xi^{(t)},\psi^{(t)}) = \left(\frac{\xi^{(t)}}{2}u^T u + \|M_+ u\|_1\right) - \left(\|M_- u\|_1 + \left(\xi^{(t)} - \psi^{(t)}\right)\|u\|_2\right) + \frac{\xi^{(t)}}{2} - \psi^{(t)},$$

we know that $L(u;\xi^{(t)},\psi^{(t)}) \to \infty$ as $\|u\|_2 \to \infty$, because the quadratic term dominates the value of $L(u;\xi^{(t)},\psi^{(t)})$. Then for any $u_0 \in \mathbb{R}^p$, the set

$$\left\{u \in \mathbb{R}^p : L(u;\xi^{(t)},\psi^{(t)}) \le L(u_0;\xi^{(t)},\psi^{(t)})\right\}$$

is bounded. $L(u_k;\xi^{(t)},\psi^{(t)})$ is also a non-increasing sequence according to Lemma 1.4.4, which indicates that for any given initial point u_0,

$$\{u_k\} \subset \left\{u \in \mathbb{R}^p : L(u;\xi^{(t)},\psi^{(t)}) \le L(u_0;\xi^{(t)},\psi^{(t)})\right\}$$

is bounded.

As $\{L(u_k;\xi^{(t)},\psi^{(t)})\}$ is bounded and also monotonically decreasing, $\{L(u_k;\xi^{(t)},\psi^{(t)})\}$ is convergent. Then, we have

$$L(u_k;\xi^{(t)},\psi^{(t)}) - L(u_{k+1};\xi^{(t)},\psi^{(t)}) \to 0 \text{ as } k \to 0.$$

From Lemma 1.4.4 we know

$$L(u_k;\xi^{(t)},\psi^{(t)}) - L(u_{k+1};\xi^{(t)},\psi^{(t)}) \ge \frac{\xi^{(t)}}{2}\|u_{k+1} - u_k\|_2^2 \ge 0,$$

so we have

$$\|u_{k+1} - u_k\|_2 \to 0 \text{ as } k \to +\infty.$$

2. Let $\{u_{k_j}\}$ be a sub-sequence of $\{u_k\}$ converging to $u^{(t)} \neq 0$.

   We know from our algorithm that

$$\begin{aligned} 0 \in\ & \xi^{(t)} u_{k_j} + M_+^T \partial \|M_+ u_{k_j}\|_1 - y_{k_j} \\ =& \xi^{(t)} u_{k_j} + M_+^T \partial \|M_+ u_{k_j}\|_1 - M_-^T \partial \|M_- u_{k_j}\|_1 - \left(\xi^{(t)} - \psi^{(t)}\right) \frac{u_{k_{j-1}}}{\|u_{k_j-1}\|_2}. \end{aligned}$$

   As $u_{k_j} \to u^{(t)}$ as $k \to \infty$, we have

$$0 \in M_+^T \partial \|M_+ u^{(t)}\|_1 - M_-^T \partial \|M_- u^{(t)}\|_1 + \xi^{(t)} u^{(t)} - \left(\xi^{(t)} - \psi^{(t)}\right) \frac{u^{(t)}}{\|u^{(t)}\|_2}.$$

3. We know that $L(u_k; \xi^{(t)}) - L(u^{(t)}; \xi^{(t)})$ can be written as

$$\begin{aligned} & L(u_k; \xi^{(t)}) - L(u^{(t)}; \xi^{(t)}) \\ = & \frac{\xi^{(t)}}{2} \left(u_k^T u_k - \left(u^{(t)}\right)^T u^{(t)}\right) + \left(\|M_+ u_k\|_1 - \|M_+ u^{(t)}\|_1\right) + \left(\|M_- u^{(t)}\|_1 - \|M_- u_k\|_1\right) \\ & + \left(\xi^{(t)} - \psi^{(t)}\right) \left(\|u^{(t)}\|_2 - \|u_k\|_2\right) \\ = & \frac{\xi^{(t)}}{2} \left(\|u_k\|_2 + \|u^{(t)}\|_2\right) \left(\|u_k\|_2 - \|u^{(t)}\|_2\right) + \left(\|M_+ u_k\|_1 - \|M_+ u^{(t)}\|_1\right) \\ & + \left(\|M_- u^{(t)}\|_1 - \|M_- u_k\|_1\right) + \left(\xi^{(t)} - \psi^{(t)}\right) \left(\|u^{(t)}\|_2 - \|u_k\|_2\right). \end{aligned}$$

   As we have proved that $\|u_k\|_2$ is bounded, suppose the upper bound is C. Then the first term in the above is upper bounded by $C\xi^{(t)} \left(\|u_k\|_2 - \|u^{(t)}\|_2\right)$. Because of the properties of the norm, we have

$$\begin{aligned} & \|u_k\|_2 - \|u^{(t)}\|_2 \leq \|u_k - u^{(t)}\|_2 \leq \|u_k - u^{(t)}\|_1, \\ & \|M_+ u_k\|_1 - \|M_+ u^{(t)}\|_1 \leq \|M_+ \left(u_k - u^{(t)}\right)\|_1 \leq |M_+| \|u_k - u^{(t)}\|_1, \\ & \|M_- u^{(t)}\|_1 - \|M_- u_k\|_1 \leq \|M_- \left(u^{(t)} - u_k\right)\|_1 \leq |M_-| \|u^{(t)} - u_k\|_1. \end{aligned}$$

Therefore, $L(u_k;\xi^{(t)}) - L(u^{(t)};\xi^{(t)})$ is upper-bounded by

$$\left(C\xi^{(t)} + |M_+| + |M_-| + \xi^{(t)} - \psi^{(t)}\right) |||u^{(t)} - u_k\|_1.$$

From Lemma 1.4.4 we know that $L(u_k;\xi^{(t)}) - L(u^{(t)};\xi^{(t)}) \geq 0$. Therefore, we can get

$$\left|L(u_k;\xi^{(t)}) - L(u^{(t)};\xi^{(t)})\right| \leq C'|||u_k - u^{(t)}\|_1,$$

where C' is a constant. Furthermore, we can get $\left|L(u_k;\xi^{(t)}) - L(u^{(t)};\xi^{(t)})\right| = O(\frac{1}{k})$.

□

APPENDIX B

PROOFS IN CHAPTER 2

All the proofs are included in this chapter, including a proof of Theorem 2.2.1 (Section B.1)), a proof of Theorem 2.3.1 (Section B.2), a proof of Theorem 2.3.2 (Section B.4), a proof of Theorem 2.3.3 (Section B.5), a proof of Lemma 2.3.4 (Section B.6), a proof of Lemma 2.3.5 (Section B.7), a proof of Lemma 2.3.7 (Section B.8), a proof of Theorem 2.3.8 (Section B.9), a proof of Lemma 2.4.1 (Section B.10), and a proof of Theorem 2.4.2 (Section B.11). Some of these proofs involves detailed and potentially tedious derivations. We try to furnish as much details as deemed reasonable.

B.1 Proof of Theorem 2.2.1

Proof. By definition of $V_{\min}$ and $V_{\max}$, we have

$$C_n V_{\min} - 1 \leq C_n \sum_{i=1}^{n} \left| u_i^T v \right| - 1 \leq C_n V_{\max} - 1.$$

The above leads to the following

$$\max_{v:||v||_2=1} \left| C_n \sum_{i=1}^{n} |u_i^T v| - 1 \right| = \max \left\{ |C_n V_{\min} - 1|, |C_n V_{\max} - 1| \right\}. \tag{1.1}$$

Consider the right hand side of the above as a function of C_n, it is verifiable that the minimum is achieved when

$$1 - C_n V_{\min} = C_n V_{\max} - 1, \text{ which leads to, } C_n = \frac{2}{V_{\min} + V_{\max}}.$$

Bringing the above to (1.1), we have

$$\left|\frac{2}{V_{\min}+V_{\max}}V_{\min}-1\right|=\frac{V_{\max}-V_{\min}}{V_{\max}+V_{\min}}=\frac{2}{1+\frac{V_{\min}}{V_{\max}}}-1. \tag{1.2}$$

From the above, it is evident that minimizing the right hand of (1.2) is equivalent to the following

$$\max_{u_1,\ldots,u_n:||u_i||_2=1}\frac{V_{\min}}{V_{\max}}.$$

From all the above, the lemma is proved. □ □

B.2 Proof of Theorem 2.3.1

Proof. Without loss of generality, we assume $\theta_i=\alpha_i+k_i\pi,$ where $\alpha_1\leq\alpha_2\leq...\leq\alpha_n\in[0,\pi)$. Then the problem in (2.9) can be written as

$$\max_{\alpha_i:i=1,\ldots,n}\frac{\min\limits_{\theta}f(\theta)}{\max\limits_{\theta}f(\theta)},$$

where $f(\theta)=\sum\limits_{i=1}^{n}|cos(\alpha_i-\theta)|$.

Let $\delta_i=\alpha_{i+1}-\alpha_i, i=1,...,n-1,$ and $\delta_n=\alpha_1-\alpha_n+\pi$. We have

$$\sum_{i=1}^{n}\delta_i=\pi.$$

For given α_i, the minimum and the maximum of $f(\theta)$ satisfy

$$\begin{aligned}
\frac{1}{n}\min_{\theta}f(\theta) &\leq \frac{1}{n}f(\alpha_i-\frac{\pi}{2}),\ \text{for } i=1,\ldots,n, &(2.3)\\
\frac{1}{n}\max_{\theta}f(\theta) &\geq \frac{1}{n}f\left(\frac{\alpha_i+\alpha_{i+1}}{2}-\frac{\pi}{2}\right),\ \text{for } i=1,\ldots,n-1, &(2.4)\\
\frac{1}{n}\max_{\theta}f(\theta) &\geq \frac{1}{n}f\left(\frac{\alpha_n+\alpha_1}{2}\right). &(2.5)
\end{aligned}$$

By summing up each side of (2.3) with i from 1 through n, we get

$$\min_{\theta} f(\theta) \leq \frac{1}{n} \sum_{i=1}^{n} f(\alpha_i - \frac{\pi}{2}). \tag{2.6}$$

By summing up each side of (2.4) with i from 1 through $n-1$ and adding it to (2.5), we have

$$\max_{\theta} f(\theta) \geq \frac{1}{n} \left[\sum_{i=1}^{n-1} f\left(\frac{\alpha_i + \alpha_{i+1}}{2} - \frac{\pi}{2} \right) + f\left(\frac{\alpha_n + \alpha_1}{2} \right) \right]. \tag{2.7}$$

Based on (2.6) and (2.7), for given α_i, we have

$$\frac{\min_{\theta} f(\theta)}{\max_{\theta} f(\theta)} \leq \frac{\frac{1}{n} \sum_{i=1}^{n} f(\alpha_i - \frac{\pi}{2})}{\frac{1}{n} \left[\sum_{i=1}^{n-1} f\left(\frac{\alpha_i + \alpha_{i+1}}{2} - \frac{\pi}{2} \right) + f\left(\frac{\alpha_n + \alpha_1}{2} \right) \right]}.$$

Therefore, one can verify the following:

$$\begin{aligned} \max_{\alpha_i : i=1,\ldots,n} \frac{\min_{\theta} f(\theta)}{\max_{\theta} f(\theta)} &\leq \max_{\alpha_i : i=1,\ldots,n} \frac{\frac{1}{n} \sum_{i=1}^{n} f(\alpha_i - \frac{\pi}{2})}{\frac{1}{n} \left[\sum_{i=1}^{n-1} f\left(\frac{\alpha_i + \alpha_{i+1}}{2} - \frac{\pi}{2} \right) + f\left(\frac{\alpha_n + \alpha_1}{2} \right) \right]} \\ &= \max_{\alpha_i : i=1,\ldots,n} \frac{\sum_{i=1}^{n} f(\alpha_i - \frac{\pi}{2})}{\left[\sum_{i=1}^{n-1} f\left(\frac{\alpha_i + \alpha_{i+1}}{2} - \frac{\pi}{2} \right) + f\left(\frac{\alpha_n + \alpha_1}{2} \right) \right]}. \end{aligned} \tag{2.8}$$

Denote the numerator of the right hand side of (2.8) as N_n, and the denominator as D_n. Thus, we have

$$N_n = \begin{cases} 2 \sum_{i=1}^{n} |\sin \delta_i| + 2 \sum_{i=1}^{n-2} |\sin(\delta_i + \delta_{i+1})| + 2 \sum_{i=1}^{n-3} |\sin(\delta_i + \delta_{i+1} + \delta_{i+2})| + ... \\ +2 \sum_{i=1}^{2} |\sin(\delta_i + \delta_{i+1} + ... + \delta_{i+n-3})|, & \text{if } n \geq 4, \\ 2 \sum_{i=1}^{n} |\sin \delta_i| & \text{if } n = 3; \end{cases}$$

and

$$\begin{aligned} D_n &= \sum_{i=1}^{n} 2\left|\sin\frac{\delta_i}{2}\right| + \sum_{i=2}^{n-1}\sum_{j=1}^{i-1}\left|\sin\left(\frac{\delta_i}{2}+\delta_{i-1}+\delta_{i-2}+...+\delta_j\right)\right| \\ &\quad + \sum_{i=1}^{n-2}\sum_{j=i+1}^{n-1}\left|\sin\left(\frac{\delta_i}{2}+\delta_{i+1}+...+\delta_j\right)\right| + \sum_{j=2}^{n-1}\left|\sin\left(\frac{\delta_n}{2}+\delta_1+...+\delta_{j-1}\right)\right|. \end{aligned}$$

We would like to show that when all the θ_i's satisfies (3.10), $\frac{\min_\theta f(\theta)}{\max_\theta f(\theta)}$ is equal to the right hand side of (2.8), which means (3.10) is the optimal solution. In order to do that, we first need to figure out what value the right hand side of (2.8) is. In the following we use perturbation analysis to show that when $\delta_i = \frac{\pi}{n}$, which is equivalent to (3.10), the right hand side achieves the maximum value. And then we show that the left side is equal to the right side under the condition of (3.10). Therefore our proof can be completed.

For $n \geq 4$, N_n and D_n are treated as functions of Δ. Then we have

$$\begin{aligned} N_n(\delta_1+\Delta, \delta_2-\Delta, \delta_3, ..., \delta_n) &= 2\left|\sin(\delta_1+\Delta)\right| + 2\left|\sin(\delta_2-\Delta)\right| \\ &\quad +2\sum_{j=3}^{n-1}\left|\sin(-\Delta+\sum_{i=2}^{j}\delta_i)\right| + Const, \end{aligned}$$

and

$$\begin{aligned} \left.\frac{\partial N_n(\delta_1+\Delta, \delta_2-\Delta, \delta_3, ..., \delta_n)}{\partial\Delta}\right|_{\Delta=0} &= 2\cos\delta_1 \mathbf{sign}(\sin\delta_1) - 2\cos\delta_2 \mathbf{sign}(\sin\delta_2) \\ &\quad -2\sum_{j=3}^{n-1}\cos\left(\sum_{i=2}^{j}\delta_i\right)\mathbf{sign}\left(\sin\left(\sum_{i=2}^{j}\delta_i\right)\right). \end{aligned}$$

When $\delta_i = \frac{\pi}{n}, i = 1, ..., n$, we have

$$\begin{aligned} \left.\frac{\partial N_n(\delta_1+\Delta, \delta_2-\Delta, \delta_3, ..., \delta_n)}{\partial\Delta}\right|_{\Delta=0} &= 0 - 2\sum_{j=3}^{n-1}\cos\left(\frac{(j-1)\pi}{n}\right)\mathbf{sign}\left(\sin\left(\frac{(j-1)\pi}{n}\right)\right) \\ &= 0. \end{aligned} \tag{2.9}$$

Similarly, for D_n, we have

$$\begin{aligned}
& D_n(\delta_1+\Delta,\delta_2-\Delta,\delta_3,...,\delta_n) \\
= \; & 2\left|\sin\left(\frac{\delta_1+\Delta}{2}\right)\right|+2\left|\sin\left(\frac{\delta_2-\Delta}{2}\right)\right|+\left|\sin\left(\frac{\Delta}{2}+\delta_1+\frac{\delta_2}{2}\right)\right| \\
& +\sum_{j=3}^{n-1}\left|\sin\left(-\Delta+\sum_{i=2}^{j-1}\delta_i+\frac{\delta_j}{2}\right)\right|+\sum_{j=3}^{n}\left|\sin\left(-\frac{1}{2}\Delta+\frac{\delta_1}{2}+\sum_{i=2}^{j-1}\delta_i\right)\right| \\
& +\sum_{j=3}^{n-1}\left|\sin\left(-\frac{1}{2}\Delta+\frac{\delta_2}{2}+\sum_{i=3}^{j}\delta_i\right)\right|+\left|\sin\left(\frac{\delta_n}{2}+\delta_1+\Delta\right)\right|+Const,
\end{aligned}$$

and

$$\begin{aligned}
& \left.\frac{\partial D_n(\delta_1+\Delta,\delta_2-\Delta,\delta_3,...,\delta_n)}{\partial\Delta}\right|_{\Delta=0} \\
= \; & \cos\frac{\delta_1}{2}\mathbf{sign}(\sin\frac{\delta_1}{2})-\cos\frac{\delta_2}{2}\mathbf{sign}(\sin\frac{\delta_2}{2})+\frac{1}{2}\cos\left(\frac{\delta_2}{2}+\delta_1\right)\mathbf{sign}\left(\sin\left(\frac{\delta_2}{2}+\delta_1\right)\right) \\
& -\sum_{j=3}^{n-1}\cos\left(\sum_{i=2}^{j-1}\delta_i+\frac{\delta_j}{2}\right)\mathbf{sign}\left(\sin\left(\sum_{i=2}^{j-1}\delta_i+\frac{\delta_j}{2}\right)\right) \\
& -\frac{1}{2}\sum_{j=2}^{n-1}\cos\left(\frac{\delta_1}{2}+\sum_{i=2}^{j}\delta_i\right)\mathbf{sign}\left(\sin\left(\frac{\delta_1}{2}+\sum_{i=2}^{j}\delta_i\right)\right) \\
& -\frac{1}{2}\sum_{j=3}^{n-1}\cos\left(\frac{\delta_2}{2}+\sum_{i=3}^{j}\delta_i\right)\mathbf{sign}\left(\sin\left(\frac{\delta_2}{2}+\sum_{i=3}^{j}\delta_i\right)\right) \\
& +\cos\left(\frac{\delta_n}{2}+\delta_1\right)\mathbf{sign}\left(\sin\left(\frac{\delta_n}{2}+\delta_1\right)\right).
\end{aligned}$$

When $\delta_i = \frac{\pi}{n}, i = 1, ..., n$, we have

$$\begin{aligned}
& \left.\frac{\partial D_n(\delta_1+\Delta, \delta_2-\Delta, \delta_3, ..., \delta_n)}{\partial \Delta}\right|_{\Delta=0} \\
= \; & 0 + \frac{1}{2}\cos\left(\frac{3\pi}{2n}\right) \text{sign}\left(\sin\left(\frac{3\pi}{2n}\right)\right) - \sum_{j=3}^{n-1} \cos\left(\frac{(2j-3)\pi}{2n}\right) \text{sign}\left(\sin\left(\frac{(2j-3)\pi}{2n}\right)\right) \\
& -\frac{1}{2}\sum_{j=2}^{n-1} \cos\left(\frac{(2j-1)\pi}{2n}\right) \text{sign}\left(\sin\left(\frac{(2j-1)\pi}{2n}\right)\right) \\
& -\frac{1}{2}\sum_{j=3}^{n-1} \cos\left(\frac{(2j-3)\pi}{2n}\right) \text{sign}\left(\sin\left(\frac{(2j-3)\pi}{2n}\right)\right) \\
& + \cos\left(\frac{3\pi}{2n}\right) \text{sign}\left(\sin\left(\frac{3\pi}{2n}\right)\right) \\
= \; & 0. \qquad (2.10)
\end{aligned}$$

Define $g(\Delta)$ as the following

$$g(\Delta) = \frac{N_n(\delta_1+\Delta, \delta_2-\Delta, \delta_3, ..., \delta_n)}{D_n(\delta_1+\Delta, \delta_2-\Delta, \delta_3, ..., \delta_n)}.$$

Then we have

$$\left.\frac{\partial g(\Delta)}{\partial \Delta}\right|_{\Delta=0} = \frac{\left.N_n'\right|_{\Delta=0}}{D_n(0)} - \frac{N_n(0)\left.D_n'\right|_{\Delta=0}}{D_n(0)^2},$$

where

$$\begin{aligned}
\left.N_n'\right|_{\Delta=0} &= \left.\frac{\partial N_n(\delta_1+\Delta, \delta_2-\Delta, \delta_3, ..., \delta_n)}{\partial \Delta}\right|_{\Delta=0}, \\
\left.D_n'\right|_{\Delta=0} &= \left.\frac{\partial D_n(\delta_1+\Delta, \delta_2-\Delta, \delta_3, ..., \delta_n)}{\partial \Delta}\right|_{\Delta=0}; \\
N_n(0) &= N_n(\delta_1, \delta_2, \delta_3, ..., \delta_n), \\
D_n(0) &= D_n(\delta_1, \delta_2, \delta_3, ..., \delta_n).
\end{aligned}$$

According to (2.9) and (2.10), we have

$$N_n'\Big|_{\Delta=0} = D_n'\Big|_{\Delta=0} = 0.$$

So we can get $\frac{\partial g(\Delta)}{\partial \Delta}\Big|_{\Delta=0} = 0 - 0 = 0.$

Similarly, for any two δ_i, δ_j, simply give some perturbation to them, we can get the same result as above. Therefore we can conclude that, for $n \geq 4$, $\left\{\delta_i = \frac{\pi}{n}, i = 1, ..., n\right\}$ can maximize the function $\frac{N_n}{D_n}$. Furthermore, we can get the maximum of $\frac{N_n}{D_n}$ by letting each δ_i be $\frac{\pi}{n}$:

$$\left(\frac{N_n}{D_n}\right)_{\max} = \frac{2n \sin\frac{\pi}{n} + 2\sum_{r=2}^{n-2}(n-r)\sin\frac{r\pi}{n}}{2n\sin\frac{\pi}{n} + \sum_{r=1}^{n-2}[2(n-r)-1]\sin\frac{(2r+1)\pi}{2n}}. \tag{2.11}$$

Next, we would like to show that when $\delta_i = \frac{\pi}{n}, i = 1, ..., n$, we have

$$\frac{\min_\theta f(\theta)}{\max_\theta f(\theta)} = \left(\frac{N_n}{D_n}\right)_{\max}.$$

As $f(\theta) = \sum_{i=1}^{n}\left|\cos\left(\theta - \frac{(i-1)\pi}{n}\right)\right|$, we know $f(\theta) = f\left(\theta - \frac{\pi}{n}\right)$. So we only need to consider $\theta \in [0, \frac{\pi}{n}]$ to get the maximum.

Recall $f(\theta)$ is linear, so the minimum and maximum must be either $\theta = 0$ or $\theta = \frac{\pi}{n}$. By observing the periodicity of the function $f(\theta)$, we can get

$$\min_\theta f(\theta) = \begin{cases} f(0) = 2\sum_{r=1}^{a-1}\sin\frac{r\pi}{2a} + 1 & \text{if } n = 2a, \\ f\left(\frac{\pi}{2(2a+1)}\right) = 2\sum_{r=1}^{a}\sin\frac{r\pi}{2a+1} & \text{if } n = 2a+1. \end{cases}$$

$$\max_\theta f(\theta) = \begin{cases} f\left(\frac{\pi}{4a}\right) = 2\sum_{r=1}^{a}\sin\frac{(2r-1)\pi}{4a} & \text{if } n = 2a. \\ f(0) = 2\sum_{r=1}^{a}\sin\frac{(2r-1)\pi}{2(2a+1)} + 1 & \text{if } n = 2a+1, \end{cases}$$

From (2.11) we can get

$$\left(\frac{N_n}{D_n}\right)_{\max} = \begin{cases} \frac{2\sum_{r=1}^{a-1}\sin\frac{r\pi}{2a}+1}{2\sum_{r=1}^{a}\sin\frac{(2r-1)\pi}{4a}} & \text{if } n = 2a. \\ \frac{2\sum_{r=1}^{a}\sin\frac{r\pi}{2a+1}}{2\sum_{r=1}^{a}\sin\frac{(2r-1)\pi}{2(2a+1)}+1} & \text{if } n = 2a+1, \end{cases} \tag{2.12}$$

Therefore, we can conclude that when $\delta_i = \frac{\pi}{n}, i = 1, ..., n$,

$$\frac{\min_\theta f(\theta)}{\max_\theta f(\theta)} = \left(\frac{N_n}{D_n}\right)_{\max}.$$

Recall the definition of δ_i's, we know that (3.10) is the optimal solution for $n \geq 4$.

For $n = 3$ and 2, by applying the similar strategy, we can get the same result as above.

□ □

B.3 Propositions we need in order to prove Theorem 2.3.2

Before proceeding to the proof of Theorem 2.3.2, we need the following Proposition B.3.1 and B.3.2:

Proposition B.3.1.

$$\sum_{s=1}^{n-1}\sin\frac{s}{n}\pi = cot\frac{\pi}{2n},$$
$$\sum_{s=1}^{n-1}\cos\frac{s}{n}\pi = 0,$$
$$\sum_{s=1}^{n-1}s\sin\frac{s}{n}\pi = \frac{n}{2}cot\frac{\pi}{2n},$$
$$\sum_{s=1}^{n-1}s\cos\frac{s}{n}\pi = -\frac{1}{2}cot^2\frac{\pi}{2n} + \frac{n-1}{2},$$
$$\sum_{s=1}^{n-1}s^2\cos\frac{s}{n}\pi = -\frac{n}{2}cot^2\frac{\pi}{2n} + \frac{n(n-1)}{2}.$$

Proof. As the following holds true

$$\sin\frac{s\pi}{N}\sin\frac{\pi}{2n}=\frac{1}{2}\left(\cos\frac{(2s-1)\pi}{2n}-\cos\frac{(2s+1)\pi}{2n}\right),$$

we have

$$\begin{aligned}&\left(\sum_{s=1}^{n-1}\sin\frac{s}{n}\pi\right)\cdot\sin\frac{\pi}{2n}\\ =\;&\frac{1}{2}\sum_{s=1}^{n-1}\left(\cos\frac{(2s-1)\pi}{2n}-\cos\frac{(2s+1)\pi}{2n}\right)=\frac{1}{2}\left(\cos\frac{\pi}{2n}-\cos\frac{(2n-1)\pi}{2n}\right)=\cos\frac{\pi}{2n}.\end{aligned}$$

So by dividing $\sin\frac{\pi}{2n}$ for both sides, we can get

$$\sum_{s=1}^{n-1}\sin\frac{s}{n}\pi=\cot\frac{\pi}{2n}. \tag{3.13}$$

As we also have

$$\cos\frac{s\pi}{N}\sin\frac{\pi}{2n}=\frac{1}{2}\left(\sin\frac{(2s+1)\pi}{2n}-\sin\frac{(2s-1)\pi}{2n}\right).$$

Therefore, we can get

$$\begin{aligned}&\left(\sum_{s=1}^{n-1}\cos\frac{s}{n}\pi\right)\cdot\sin\frac{\pi}{2n}\\ =\;&\frac{1}{2}\sum_{s=1}^{n-1}\left(\sin\frac{(2s+1)\pi}{2n}-\sin\frac{(2s-1)\pi}{2n}\right)=\frac{1}{2}\left(\sin\frac{(2n-1)\pi}{2n}-\sin\frac{\pi}{2n}\right)=0,\end{aligned}$$

which implies

$$\sum_{s=1}^{n-1}\cos\frac{s}{n}\pi=0. \tag{3.14}$$

As we also have

$$\sin\frac{s}{n}\pi\cdot\sin\frac{\pi}{2n}=\cos\frac{(2s-1)\pi}{2n}-\cos\frac{(2s+1)\pi}{2n},$$

the following can be derived:

$$\left(\sum_{s=1}^{n-1} s \sin\frac{s}{n}\pi\right) \cdot \sin\frac{\pi}{2n} = \frac{1}{2}\sum_{s=1}^{n-1} s \cdot \left(\cos\frac{(2s-1)\pi}{2n} - \cos\frac{(2s+1)\pi}{2n}\right). \tag{3.15}$$

Since we have

$$\begin{aligned} & \sum_{s=1}^{n-1} s \cdot \left(\cos\frac{(2s-1)\pi}{2n} - \cos\frac{(2s+1)\pi}{2n}\right) \\ = & \sum_{s=1}^{n-1} \cos\frac{(2s-1)\pi}{2n} - (n-1)\cos\frac{(2n-1)\pi}{2n} \\ = & \sum_{s=1}^{n-1} \left(\cos\frac{s}{n}\pi\cos\frac{\pi}{2n} + \sin\frac{s}{n}\pi\sin\frac{\pi}{2n}\right) + (n-1)\cos\frac{\pi}{2n}, \end{aligned}$$

by plugging the above as well as (3.13) and (3.14) into (3.15), we can get

$$\begin{aligned} & \left(\sum_{s=1}^{n-1} s \sin\frac{s}{n}\pi\right) \cdot \sin\frac{\pi}{2n} \\ = & \frac{1}{2}\left(\sum_{s=1}^{n-1} \left(\cos\frac{s}{n}\pi\cos\frac{\pi}{2n} + \sin\frac{s}{n}\pi\sin\frac{\pi}{2n}\right) + (n-1)\cos\frac{\pi}{2n}\right) \\ = & \frac{1}{2}\left(0 + \cos\frac{\pi}{2n}\right) + \frac{n-1}{2}\cos\frac{\pi}{2n} = \frac{n}{2}\cos\frac{\pi}{2n}. \end{aligned} \tag{3.16}$$

Similarly, since we have

$$\cos\frac{s}{n}\pi \cdot \sin\frac{\pi}{2n} = \sin\frac{(2s+1)\pi}{2n} - \sin\frac{(2s-1)\pi}{2n},$$

by using the similar strategy, we can get

$$\left(\sum_{s=1}^{n-1} s \cos\frac{s}{n}\pi\right) \cdot \sin\frac{\pi}{2n} = -\frac{1}{2}\cos\frac{\pi}{2n}\cot\frac{\pi}{2n} + \frac{n-1}{2}\sin\frac{\pi}{2n}. \tag{3.17}$$

Therefore, dividing both the equations (3.16) and (3.17), we can get

$$\sum_{s=1}^{n-1} s \sin \frac{s}{n}\pi = \frac{n}{2}\cot\frac{\pi}{2n}, \tag{3.18}$$

$$\sum_{s=1}^{n-1} s \cos \frac{s}{n}\pi = -\frac{1}{2}\cot^2\frac{\pi}{2n} + \frac{n-1}{2}. \tag{3.19}$$

Since the following holds true,

$$\left(\sum_{s=1}^{n-1} s^2 \cos \frac{s}{n}\pi\right) \cdot \sin\frac{\pi}{2n} = \frac{1}{2}\left\{-\sum_{s=1}^{n-1}(2s-1)\sin\frac{(2s-1)\pi}{2n} + (n-1)^2 \sin\frac{2n-1}{2n}\pi\right\},$$

by simplifying the above equation we can get

$$\begin{aligned}
&\left(\sum_{s=1}^{n-1} s^2 \cos \frac{s}{n}\pi\right) \cdot \sin\frac{\pi}{2n} \\
= \; & -\sum_{s=1}^{n-1} s\left(\sin\frac{s}{n}\pi\cos\frac{\pi}{2n} - \cos\frac{s}{n}\pi\sin\frac{\pi}{2n}\right) + \frac{1}{2}\sum_{s=1}^{n-1}\left(\sin\frac{s}{n}\pi\cos\frac{\pi}{2n} - \cos\frac{s}{n}\pi\sin\frac{\pi}{2n}\right) \\
& + \frac{(n-1)^2}{2}\sin\frac{\pi}{2n} \\
= \; & -\left(\sum_{s=1}^{n-1} s\sin\frac{s}{n}\pi\right)\cos\frac{\pi}{2n} + \left(\sum_{s=1}^{n-1} s\cos\frac{s}{n}\pi\right)\sin\frac{\pi}{2n} + \frac{1}{2}\left(\sum_{s=1}^{n-1}\sin\frac{s}{n}\pi\right)\cos\frac{\pi}{2n} \\
& - \frac{1}{2}\left(\sum_{s=1}^{n-1}\cos\frac{s}{n}\pi\right)\sin\frac{\pi}{2n} + \frac{(n-1)^2}{2}\sin\frac{\pi}{2n}.
\end{aligned}$$

Plugging (3.13), (3.14), (3.18), and (3.19) into the above, we can get

$$\left(\sum_{s=1}^{n-1} s^2 \cos \frac{s}{n}\pi\right) \cdot \sin\frac{\pi}{2n} = -\frac{1}{2}\cot^2\frac{\pi}{2n}\sin\frac{\pi}{2n} + \frac{n(n-1)}{2}\sin\frac{\pi}{2n}.$$

Therefore, dividing $\sin\frac{\pi}{2n}$ on each side, we get

$$\sum_{s=1}^{n-1} s^2 \cos \frac{s}{n}\pi = -\frac{n}{2}\cot^2\frac{\pi}{2n} + \frac{n(n-1)}{2}.$$

□ □

Proposition B.3.2.

$$2\sum_{s=1}^{n-1}(n-s)f(s) = \frac{n}{\pi}cot\frac{\pi}{2n} + \frac{1}{2}cot^2\frac{\pi}{2n} - \frac{n}{2} + \frac{1}{2}.$$

Proof. According to the definition of function $f(s)$ in (4.22), we have

$$\begin{aligned} & 2\sum_{s=1}^{n-1}(n-s)f(s) \\ = \; & 2\sum_{s=1}^{n-1}(n-s)\left(\frac{1}{\pi}\sin\frac{s}{n}\pi + \left(\frac{1}{2} - \frac{s}{n}\right)\cos\frac{s}{n}\pi\right) \\ = \; & \frac{2n}{\pi}\sum_{s=1}^{n-1}\sin\frac{s}{n}\pi - \frac{2}{\pi}\sum_{s=1}^{n-1}s\sin\frac{s}{n}\pi + n\sum_{s=1}^{n-1}\cos\frac{s}{n}\pi - 3\sum_{s=1}^{n-1}s\cos\frac{s}{n}\pi + \frac{2}{n}\sum_{s=1}^{n-1}s^2\cos\frac{s}{n}\pi. \end{aligned}$$

Applying Proposition B.3.1,we have

$$2\sum_{s=1}^{n-1}(n-s)f(s) = \frac{n}{\pi}\cot\frac{\pi}{2n} + \frac{1}{2}\cot^2\frac{\pi}{2n} - \frac{n}{2} + \frac{1}{2}. \qquad \square$$

□

B.4 Proof of Theorem 2.3.2

Proof. Recall that u_i can be rewritten as

$$u_i = e^{\sqrt{-1}\frac{i\pi}{n}}, i = 0, 1, \ldots, n-1.$$

And we have

$$\begin{aligned}
&\mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left\{\left|C_n\sum_{i=1}^{n}\left|u_i^Tv\right|-1\right|^2\right\}\\
=\;& C_n^2\mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left\{\left(\sum_{i=1}^{n}\left|u_i^Tv\right|\right)^2\right\}-2C_n\mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left\{\sum_{i=1}^{n}\left|u_i^Tv\right|\right\}+1\\
=\;& C_n^2\sum_{i=1}^{n}\mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left(\left|u_i^Tv\right|^2\right)+2C_n^2\sum_{1\le i<j\le N}\mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left(\left|u_i^Tv\right|\left|u_j^Tv\right|\right)\\
&-2C_n\sum_{i=1}^{n}\mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left\{\left|u_i^Tv\right|\right\}+1. \qquad (4.20)
\end{aligned}$$

So we will find out the expected squared error, if for all $i,j=1,\ldots,n$, we can get the values of

$$\mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left(\left|u_i^Tv\right|^2\right),\quad \mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left(\left|u_i^Tv\right|\left|u_j^Tv\right|\right),\quad \mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left\{\left|u_i^Tv\right|\right\}.$$

In order to calculate $\mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left(\left|u_i^Tv\right|^2\right)$, we let $u_i=(1,0)'$ and $v=(\cos\theta,\sin\theta)'$ without loss of generality. Then,

$$\mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left(\left|u_i^Tv\right|^2\right)=\mathop{\mathbb{E}}_{\theta\sim\text{Unif}(0,2\pi)}\cos^2\theta=\frac{1}{2}+\frac{1}{2}\mathop{\mathbb{E}}_{\theta\sim\text{Unif}(0,2\pi)}\cos 2\theta=\frac{1}{2}.$$

Without loss of generality, assume $\langle u_i,u_j\rangle=\frac{s}{n}\pi,$ for all $1\le i,j\le n, i\ne j$, which means we can assume

$$u_i=(1,0)', u_j=(\cos\frac{s}{n}\pi,\sin\frac{s}{n}\pi)', s=1,2,\ldots,n-1.$$

Therefore, we have

$$\begin{aligned}
\left|u_i^Tv\right|\cdot\left|u_j^Tv\right| &= |\cos\theta|\left|\cos\theta\cos\frac{s}{n}\pi+\sin\theta\sin\frac{s}{n}\pi\right|\\
&= \left|\cos^2\theta\cos\frac{s}{n}\pi+\cos\theta\sin\theta\sin\frac{s}{n}\pi\right|.
\end{aligned}$$

As the following equations hold,

$$\cos^2\theta = \frac{1+\cos 2\theta}{2} \text{ and } \cos\theta\sin\theta = \frac{\sin 2\theta}{2},$$

quantity $\left|u_i^T v\right| \cdot \left|u_j^T v\right|$ can be further written as

$$\begin{aligned}\left|u_i^T v\right| \cdot \left|u_j^T v\right| &= \frac{1}{2}\left|\cos 2\theta \cos\frac{s}{n}\pi + \sin 2\theta \sin\frac{s}{n}\pi + \cos\frac{s}{n}\pi\right| \\ &= \frac{1}{2}\left|\cos\left(2\theta - \frac{s}{n}\pi\right) + \cos\frac{s}{n}\pi\right|.\end{aligned}$$

So $\mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left(\left|u_i^T v\right| \cdot \left|u_j^T v\right|\right)$ can be rewritten as follows:

$$\begin{aligned}&\mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left(\left|u_i^T v\right| \cdot \left|u_j^T v\right|\right) \\ =\ & \frac{1}{2}\mathop{\mathbb{E}}_{\theta\sim\text{Unif}(0,2\pi)}\left\{\left|\cos\left(2\theta - \frac{s}{n}\pi\right) + \cos\frac{s}{n}\pi\right|\right\} \\ =\ & \frac{1}{2}\times\frac{1}{2\pi}\left(\int_0^{\pi} + \int_{\pi}^{2\pi}\right)\left|\cos\left(2\theta - \frac{s}{n}\pi\right) + \cos\frac{s}{n}\pi\right| \mathrm{d}\theta.\end{aligned}$$

As we have

$$\begin{aligned}&\int_{\pi}^{2\pi}\left|\cos\left(2\theta - \frac{s}{n}\pi\right) + \cos\frac{s}{n}\pi\right| \mathrm{d}\theta \\ =\ & \int_0^{\pi}\left|\cos\left(2\theta - \frac{s}{n}\pi\right) + \cos\frac{s}{n}\pi\right| \mathrm{d}\theta = \int_0^{\pi}\left|\cos\left(2\theta\right) + \cos\frac{s}{n}\pi\right| d\theta \\ =\ & \int_{-\frac{\pi}{2}+\frac{s}{2n}\pi}^{\frac{\pi}{2}+\frac{s}{2n}\pi}\left|\cos\left(2\theta\right) + \cos\frac{s}{n}\pi\right| d\theta,\end{aligned}$$

we can get

$$\mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left(\left|u_i^T v\right| \cdot \left|u_j^T v\right|\right) = \frac{1}{2\pi}\int_{-\frac{\pi}{2}+\frac{s}{2n}\pi}^{\frac{\pi}{2}-\frac{s}{2n}\pi}\left|\cos\left(2\theta\right) + \cos\frac{s}{n}\pi\right| d\theta. \tag{4.21}$$

By breaking the integral interval $(-\frac{\pi}{2}+\frac{s}{2n}\pi, \frac{\pi}{2}+\frac{s}{2n}\pi)$ into two sub-intervals, $(-\frac{\pi}{2}+\frac{s}{2n}\pi, \frac{\pi}{2}-\frac{s}{2n}\pi)$ and $(\frac{\pi}{2}-\frac{s}{2n}\pi, \frac{\pi}{2}+\frac{s}{2n}\pi)$, we have

$$\left|\cos 2\theta + \cos\frac{s}{n}\pi\right| = \begin{cases} \cos 2\theta + \cos\frac{s}{n}\pi, & \theta \in (-\frac{\pi}{2}+\frac{s}{2n}\pi, \frac{\pi}{2}-\frac{s}{2n}\pi), \\ -\left(\cos 2\theta + \cos\frac{s}{n}\pi\right), & \theta \in (\frac{\pi}{2}-\frac{s}{2n}\pi, \frac{\pi}{2}+\frac{s}{2n}\pi). \end{cases}$$

Combining (4.21), we get

$$\begin{aligned} & \mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)} \left(\left|u_i^T v\right| \cdot \left|u_j^T v\right|\right) \\ = & \frac{1}{2\pi}\left(\int_{-\frac{\pi}{2}+\frac{s}{2n}\pi}^{\frac{\pi}{2}-\frac{s}{2n}\pi} + \int_{\frac{\pi}{2}-\frac{s}{2n}\pi}^{\frac{\pi}{2}+\frac{s}{2n}\pi}\right)\left|\cos 2\theta + \cos\frac{s}{n}\pi\right| d\theta \\ = & \frac{1}{2\pi}\left\{\int_{-\frac{\pi}{2}+\frac{s}{2n}\pi}^{\frac{\pi}{2}-\frac{s}{2n}\pi}\left(\cos 2\theta + \cos\frac{s}{n}\pi\right) d\theta - \int_{\frac{\pi}{2}-\frac{s}{2n}\pi}^{\frac{\pi}{2}+\frac{s}{2n}\pi}\left(\cos 2\theta + \cos\frac{s}{n}\pi\right) d\theta\right\} \\ = & \frac{1}{2\pi}\left\{2\sin\frac{s}{n}\pi + \left(\pi - \frac{2s}{N}\pi\right)\cos\frac{s}{n}\pi\right\} \\ = & \frac{1}{\pi}\sin\frac{s}{n}\pi + \left(\frac{1}{2}-\frac{s}{n}\right)\cos\frac{s}{n}\pi. \end{aligned}$$

If we define

$$f(s) = \frac{1}{\pi}\sin\frac{s}{n}\pi + \left(\frac{1}{2}-\frac{s}{n}\right)\cos\frac{s}{n}\pi, s = 0, 1, 2, \cdots, n-1. \tag{4.22}$$

Then we will get

$$\begin{aligned} \mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)} \left(\left|u_i^T v\right|^2\right) &= f(0), \\ \mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)} \left(\left|u_i^T v\right| \cdot \left|u_j^T v\right|\right) &= f(s), \text{ where } \langle u_i, u_j\rangle = \frac{s}{n}\pi, s = 1, 2, \cdots, n-1 \end{aligned} \tag{4.23}$$

Similarly, without loss of generality, if we assume $u_i = (1, 0)', v = (\cos\theta, \sin\theta)'$, the

following holds,

$$\mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left\{\left|u_i^T v\right|\right\} = \mathop{\mathbb{E}}_{\theta\sim\text{Unif}(-\pi,\pi)}|\cos\theta| = 2\int_{-\frac{\pi}{2}}^{\frac{\pi}{2}}\frac{1}{2\pi}\cos\theta d\theta = \frac{2}{\pi}. \tag{4.24}$$

Recall that we have

$$C_n = \frac{2}{\mathrm{V}_{\min}+\mathrm{V}_{\max}},\ \text{where}\ \mathrm{V}_{\min} = \min_{v:\|v\|=1}\sum_{i=1}^{n}\left|u_i^T v\right|, \mathrm{V}_{\max} = \max_{v:\|v\|=1}\sum_{i=1}^{n}\left|u_i^T v\right|.$$

From (2.12) we can easily verify that

$$\mathrm{V}_{\min}+\mathrm{V}_{\max} = 2\sum_{k=1}^{n-1}\sin\frac{k\pi}{2n}+1.$$

Therefore, C_n can be derived:

$$C_n = \frac{2}{2\sum_{k=1}^{n-1}\sin\frac{k\pi}{2n}+1}. \tag{4.25}$$

As we have

$$\sin\frac{k\pi}{2n}\cdot\sin\frac{\pi}{4n} = \frac{1}{2}\left(\cos\frac{(2k-1)\pi}{4n}-\cos\frac{(2k+1)\pi}{4n}\right),$$

we can get

$$\sin\frac{\pi}{4n}\left(\sum_{k=1}^{n-1}\sin\frac{k\pi}{2n}\right) = \frac{1}{2}\sum_{k=1}^{n-1}\left(\cos\frac{(2k-1)\pi}{4n}-\cos\frac{(2k+1)\pi}{4n}\right) = \frac{1}{2}\left(\cos\frac{\pi}{4n}-\sin\frac{\pi}{4n}\right),$$

which leads to

$$\sum_{k=1}^{n-1}\sin\frac{k\pi}{2n} = \frac{\frac{1}{2}\left(\cos\frac{\pi}{4n}-\sin\frac{\pi}{4n}\right)}{\sin\frac{\pi}{4n}} = \frac{1}{2}\cot\frac{\pi}{4n}-\frac{1}{2}. \tag{4.26}$$

Therefore, by plugging (4.26) into (4.25), we have

$$C_n = \frac{2}{\cot\frac{\pi}{4n}} = 2\tan\frac{\pi}{4n}.$$

If we plug in (4.20) with (4.23) and (4.24), we can get

$$\begin{aligned}
&\mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left\{\left|C_n\sum_{i=1}^{n}\left|u_i^Tv\right|-1\right|^2\right\}\\
=\;& C_n^2\sum_{i=1}^{n}\mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left(\left|u_i^Tv\right|^2\right)+2C_n^2\sum_{1\le i<j\le N}\mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left(\left|u_i^Tv\right|\left|u_j^Tv\right|\right)\\
&-2C_n\sum_{i=1}^{n}\mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left\{\left|u_i^Tv\right|\right\}+1\\
=\;& 4\tan^2\frac{\pi}{4n}\left(\frac{n}{2}+2\sum_{s=1}^{n-1}(n-s)f(s)\right)-\frac{8N}{\pi}\tan\frac{\pi}{4n}+1. \qquad (4.27)
\end{aligned}$$

In order to calculate the part $\sum_{s=1}^{n-1}(n-s)f(s)$ in (4.27), we need the Proposition B.3.2. Applying Proposition B.3.2 on (4.27), we get

$$\begin{aligned}
&\mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left\{\left|C_n\sum_{i=1}^{n}\left|u_i^Tv\right|-1\right|^2\right\}\\
=\;& 4\tan^2\frac{\pi}{4n}\left(\frac{n}{2}+\frac{n}{\pi}\cot\frac{\pi}{2n}+\frac{1}{2}\cot^2\frac{\pi}{2n}-\frac{n}{2}+\frac{1}{2}\right)-\frac{8n}{\pi}\tan\frac{\pi}{4n}+1\\
=\;& 2\tan^2\frac{\pi}{4n}\cot^2\frac{\pi}{2n}+\frac{4n}{\pi}\tan^2\frac{\pi}{4n}\cot\frac{\pi}{2n}+2\tan^2\frac{\pi}{4n}-\frac{8n}{\pi}\tan\frac{\pi}{4n}+1. \qquad (4.28)
\end{aligned}$$

As $\tan x \to x,\ \text{as } x\to 0$, we can get

$$\begin{aligned}
\mathop{\mathbb{E}}_{v\sim\text{Unif}(S^1)}\left\{\left|C_n\sum_{i=1}^{n}\left|u_i^Tv\right|-1\right|^2\right\} \;&\longrightarrow\; 2\frac{\pi^2}{16n^2}\frac{4n^2}{\pi^2}+\frac{4n}{\pi}\frac{\pi^2}{16n^2}\frac{2n}{\pi}+2\frac{\pi^2}{16n^2}-\frac{8n}{\pi}\frac{\pi}{4n}+1\\
&= \frac{\pi^2}{8n^2}. \qquad \square
\end{aligned}$$

□

B.5 Proof of Theorem 2.3.3

Proof. Monte Carlo method uses random directions to approximate the norm, which means

$$u_i \sim \text{Unif}(S^1), i.i.d.$$

We also know that

$$\begin{aligned}
&\mathop{\mathbb{E}}_{u_i,v\sim\text{Unif}(S^1)} \left\{ \left| C_n \sum_{i=1}^{n} \left|u_i^T v\right| - 1 \right|^2 \right\} \\
= \; & C_n^2 \mathop{\mathbb{E}}_{u_i,v\sim\text{Unif}(S^1)} \left\{ \left(\sum_{i=1}^{n} \left|u_i^T v\right| \right)^2 \right\} - 2C_n \mathop{\mathbb{E}}_{u_i,v\sim\text{Unif}(S^1)} \left\{ \sum_{i=1}^{n} \left|u_i^T v\right| \right\} + 1 \\
= \; & C_n^2 \sum_{i=1}^{n} \mathop{\mathbb{E}}_{u_i,v\sim\text{Unif}(S^1)} \left(\left|u_i^T v\right|^2 \right) + 2C_n^2 \sum_{1\le i<j\le N} \mathop{\mathbb{E}}_{u_i,u_j,v\sim\text{Unif}(S^1)} \left(\left|u_i^T v\right| \left|u_j^T v\right| \right) \\
& -2C_n \sum_{i=1}^{n} \mathop{\mathbb{E}}_{u_i,v\sim\text{Unif}(S^1)} \left\{ \left|u_i^T v\right| \right\} + 1, \qquad (5.29)
\end{aligned}$$

where C_n satisfies

$$C_n \cdot \int_{u_i\in S^1} \sum_{i=1}^{n} \left|u_i^T v\right| du_i = 1,$$

which implies

$$C_n = \frac{\pi}{2n}.$$

We can find out the expected squared error if we can get the values of

$$\begin{aligned}
&\mathop{\mathbb{E}}_{u_i,v\sim\text{Unif}(S^1)} \left(\left|u_i^T v\right|^2 \right), \\
&\mathop{\mathbb{E}}_{u_i,u_j,v\sim\text{Unif}(S^1)} \left(\left|u_i^T v\right| \left|u_j^T v\right| \right), \\
&\mathop{\mathbb{E}}_{u_i,v\sim\text{Unif}(S^1)} \left\{ \left|u_i^T v\right| \right\}, \quad \text{for all } i,j = 1,\cdots,n.
\end{aligned}$$

Let $u_i = (\cos\phi, \sin\phi)', v = (\cos\theta, \sin\theta)'$, where $\phi \sim \text{Unif}(0, 2\pi), \theta \sim \text{Unif}(0, 2\pi)$. Then the above three can be computed as follows:

$$\begin{aligned}
&\mathop{\mathbb{E}}_{u_i, v \sim \text{Unif}(S^1)} \left(\left| u_i^T v \right|^2 \right) \\
= &\mathop{\mathbb{E}}_{\phi, \theta \sim \text{Unif}(0,2\pi)} \cos^2(\phi - \theta) = \mathop{\mathbb{E}}_{\phi \sim \text{Unif}(0,2\pi)} \left[\mathop{\mathbb{E}}_{\theta \sim \text{Unif}(0,2\pi)} \left[\cos^2(\phi - \theta) | \phi \right] \right] \\
= &\frac{1}{2},
\end{aligned}$$

and

$$\begin{aligned}
&\mathop{\mathbb{E}}_{u_i, u_j, v \sim \text{Unif}(S^1)} \left(\left| u_i^T v \right| \left| u_j^T v \right| \right) \\
= &\mathop{\mathbb{E}}_{\phi_i, \phi_j, \theta \sim \text{Unif}(0,2\pi)} \left\{ \left| \cos(\theta - \phi_i) \right| \left| \cos(\theta - \phi_j) \right| \right\} \\
= &\mathop{\mathbb{E}}_{\phi_j, \theta \sim \text{Unif}(0,2\pi)} \left\{ \mathop{\mathbb{E}}_{\phi_i \sim \text{Unif}(0,2\pi)} \left[\left| \cos(\theta - \phi_i) \right| \left[\left| \cos(\theta - \phi_j) \right| | \phi_j, \theta \right] \right\} \right. \\
= &\mathop{\mathbb{E}}_{\phi_j, \theta \sim \text{Unif}(0,2\pi)} \left\{ \left| \cos(\theta - \phi_j) \right| \cdot \frac{2}{\pi} \right\} \\
= &\mathop{\mathbb{E}}_{\theta \sim \text{Unif}(0,2\pi)} \left\{ \mathop{\mathbb{E}}_{\phi_j \sim \text{Unif}(0,2\pi)} \left[\left| \cos(\theta - \phi_j) \right| \cdot \frac{2}{\pi} | \theta \right] \right\} = \mathop{\mathbb{E}}_{\theta \sim \text{Unif}(0,2\pi)} \left[\frac{2}{\pi} \cdot \frac{2}{\pi} \right] \\
= &\frac{4}{\pi^2},
\end{aligned}$$

and

$$\begin{aligned}
&\mathop{\mathbb{E}}_{u_i, v \sim \text{Unif}(S^1)} \left\{ \left| u_i^T v \right| \right\} \\
= &\mathop{\mathbb{E}}_{\phi, \theta \sim \text{Unif}(0,2\pi)} \left| \cos(\phi - \theta) \right| = \mathop{\mathbb{E}}_{\phi \sim \text{Unif}(0,2\pi)} \left[\mathop{\mathbb{E}}_{\theta \sim \text{Unif}(0,2\pi)} \left[\left| \cos(\phi - \theta) \right| | \phi \right] \right] \\
= &\mathop{\mathbb{E}}_{\phi \sim \text{Unif}(0,2\pi)} \frac{2}{\pi} = \frac{2}{\pi}.
\end{aligned}$$

Therefore by plugging the above results into (5.29), we eventually get

$$\begin{aligned}
&\mathop{\mathbb{E}}_{u_i,v\sim\text{Unif}(S^1)} \left\{ \left| C_n \sum_{i=1}^{n} \left| u_i^T v \right| - 1 \right|^2 \right\} \\
= \quad & \frac{\pi^2}{4N^2} \left(N \cdot \frac{1}{2} + 2\frac{N(N-1)}{2} \frac{4}{\pi^2} \right) - 2\frac{\pi}{2n} \cdot N \cdot \frac{2}{\pi} + 1 = \frac{\pi^2 - 8}{8N}. \qquad \square
\end{aligned}$$

$\square$

B.6 Proof of Lemma 2.3.4

Proof. Recall that we have

$$V_{\max} = \max_{v:||v||_2=1} \sum_{i=1}^{n} \left| u_i^T v \right| = \max_{v:||v||_2=1} \max_{s_i\in\{1,-1\}} \left(\sum_{i=1}^{n} s_i u_i^T \right) v, \tag{6.30}$$

where the second equality is based on a standard trick in optimization [39, Chapter 9.2(ii)].

The following is an application of the Cauchy-Schwartz inequality:

$$\left(\sum_{i=1}^{n} s_i u_i^T \right) v \leq \sqrt{\left\| \sum_{i=1}^{n} s_i u_i \right\|_2^2 ||v||_2^2} = \left\| \sum_{i=1}^{n} s_i u_i \right\|,$$

where the equality is due to the condition $\|v\| = 1$.

In the first part, the equality holds if and only if $|v_j| = c \left| \left(\sum_{i=1}^{n} s_i u_i \right)_j \right|, j = 1, ..., p.$

Apparently, we must have $c = \left\| \sum_{i=1}^{n} s_i u_i \right\|^{-1}$ (because of $\|v\| = 1$).

So we can have

$$v = \frac{\sum_{i=1}^{n} s_i u_i}{\left\| \sum_{i=1}^{n} s_i u_i \right\|}. \tag{6.31}$$

Combining (6.31) and (6.30), we have (3.11). $\square$ $\square$

B.7 Proof of Lemma 2.3.5

Proof. We start with a special case: the linear subspace is $\mathbb{R}^p$(the entire space). Obviously the n hyperplanes

$$\left\{y : u_i^T y = 0\right\}, \text{for } i = 1, 2, ..., n$$

divide the sphere S^{p-1} into at most 2^n sectors. Within each sector, function $f(v)$ is strictly linear, therefore the minima cannot be an interior point. Recall a boundary point v must have $u_j^T v = 0$ for at least one $j, 1 \leq j \leq n$.

Now we consider a linear subspace with dimension less than p, say, k. Let $b_1, ..., b_k$ be the orthonormal basis of such a linear subspace, we have $\forall x \in \Omega$,

$$x = \sum_{j=1}^{k} c_j b_j,$$

and

$$\sum_{j=1}^{k} c_j^2 = 1, \text{(Because we have } \|x\| = 1).$$

Therefore, we have

$$f(v) = \sum_{i=1}^{n} \left|u_i^T v\right| = \sum_{i=1}^{n} \left|u_i^T \sum_{j=1}^{k} c_j b_j\right| = \sum_{i=1}^{n} \left|\sum_{j=1}^{k} c_j (u_i^T b_j)\right| = \sum_{i=1}^{n} \left|h_i^T c\right|,$$

where $c = (c_1, ..., c_k)^T$ and $h_i^T = \left(u_i^T b_1, ..., u_i^T b_k\right), i = 1, ..., n$. Note that in the early part of this proof, the u_i can be arbitrary.

The above derivation indicates that the latter case can be converted into the former case, as $c \in \mathbb{R}^k$ is from the entire space. So we can get

$$h_i^T c = 0 \text{ for at least one } i, 1 \leq i \leq n.$$

As $h_i^T c = u_i^T \left(\sum_{j=1}^{k} b_j c_j \right)$, the above is equivalent to

$$u_i^T \left(\sum_{j=1}^{k} b_j c_j \right) = 0 \text{ for at least one } i, 1 \leq i \leq n.$$

Quantity $\sum_{j=1}^{k} b_j c_j$ can also be denoted as v, because any vector on the space is a linear combination of the orthonormal basis $b_1, ..., b_k$.

From all the above, we proved the lemma. □ □

B.8 Proof of Lemma 2.3.7

Proof. For notational simplicity, let us donate $\Omega = \Omega(v_{\min})$. We can easily verify the following

$$\text{rank}(\Omega) \leq p - 1.$$

Otherwise (i.e., $\text{rank}(\Omega) = p$), by the definition of Ω, we will have $v_{\min} = 0$. Now we show that

$$\text{rank}(\Omega) \geq p - 1.$$

We use contradiction. Let us assume that $\text{rank}(\Omega) < p - 1$. Define the following complementary set

$$\Omega^{\perp} = \{x : \|x\| = 1, x \perp \Omega\},$$

where $x \perp \Omega$ stands for that x is perpendicular to the linear space that is spanned by all the u_j's in Ω. Because $v_{\min}$ is a minimizer, we have that

$$f(v_{\min}) = \min_{v \in \Omega^{\perp}} f(v) = \min_{v \in \Omega^{\perp}} \sum_{i=1}^{n} \left| u_i^T v \right| = \min_{v \in \Omega^{\perp}} \sum_{u_i \notin \Omega} \left| u_i^T v \right|$$

Note that if $\text{rank}(\Omega) < p - 1$, we have $\dim(\Omega^{\perp}) \geq 2$.

By Lemma 2.3.5, we can declare that there exists $u_j \notin \Omega$, $u_j^T v_{\min} = 0$. However, this

contradicts to the definition of Ω, which is supposed to be the maximal subset. □ □

B.9 Proof of Theorem 2.3.8

Proof. When $n = p$, we have

$$f(v) = |u_1^T v| + |u_2^T v| + ... + |u_p^T v|, \text{ for } u_1, ...u_p, v \in \mathbb{S}^{p-1}.$$

According to the Lemma 2.3.7, we have

$$\text{rank}\left(\Omega(v_{\min})\right) = p - 1,$$

where $\Omega(v_{\min}) = \left\{u_j : u_j^T v_{\min} = 0\right\}$, and $v_{\min}$ is the minimizer of $f(v)$. So the minimizer of $f(v)$ must satisfy that it is orthogonal to $p-1$ linearly independent u_j's.

Assume every $p-1$ u_j's are linearly independent. Then the minimizer is among the vectors that are orthogonal to any $p-1$ u_j's. We know there are $\binom{p}{p-1} = p$ different combinations of u_j's, and each combination is correspond to 2 unit vectors orthogonal to one of the $p-1$ u_j's. (These 2 unit vectors are the two directions that are orthogonal to a $p-1$ spaces in $\mathbb{R}^p$.) Thus there are totally $2p$ unit vectors that might be the minimizer of $f(v)$.

Suppose p of the $2p$ unit vectors are those whose first nonzero entry is positive. Denote them as $v^{-(1)}, v^{-(2)}, ..., v^{-(p)}$. Then the other p unit vectors would be $-v^{-(1)}, -v^{-(2)}, ..., -v^{-(p)}$. Suppose that for any $i \in \{1, 2, ..., p\}$, $v^{-(1)}, v^{-(2)}, ..., v^{-(p)}$ satisfy

$$\left(v^{-(i)}\right)^T u_j = 0, \forall j \neq i, j \in \{1, 2, ..., p\}.$$

Thus the minimum value of $f(v)$ can be upper bounded by the average of the function

values of the p unit vectors:

$$\min_v f(v) \leq \frac{1}{p}\sum_{i=1}^{p} f(v^{-(i)}). \tag{9.32}$$

We can also bound the maximum value of $f(v)$ by some value:

$$\max_v f(v) \geq \max_{s_i=\pm 1} f\left(\frac{\sum_{i=1}^{p} s_i v^{-(i)}}{\|\sum_{i=1}^{p} s_i v^{-(i)}\|}\right). \tag{9.33}$$

Because we have

$$f\left(\frac{\sum_{i=1}^{p} s_i v^{-(i)}}{\|\sum_{i=1}^{p} s_i v^{-(i)}\|}\right) = \frac{f\left(\sum_{i=1}^{p} s_i v^{-(i)}\right)}{\|\sum_{i=1}^{p} s_i v^{-(i)}\|},$$

and

$$\begin{aligned} f\left(\sum_{i=1}^{p} s_i v^{-(i)}\right) &= \sum_{j=1}^{p}\left|u_j^T\left(\sum_{i=1}^{p} s_i v^{-(i)}\right)\right| = \sum_{j=1}^{p}\left|\sum_{i=1}^{p} s_i u_j^T v^{-(i)}\right| = \sum_{j=1}^{p}\left|s_j u_j^T v^{-(j)}\right| \\ &= \sum_{j=1}^{p}\left|u_j^T v^{-(j)}\right| = \sum_{j=1}^{p} f\left(v^{-(j)}\right), \end{aligned}$$

we can get

$$f\left(\frac{\sum_{i=1}^{p} s_i v^{-(i)}}{\|\sum_{i=1}^{p} s_i v^{-(i)}\|}\right) = \frac{\sum_{j=1}^{p} f\left(v^{-(j)}\right)}{\|\sum_{i=1}^{p} s_i v^{-(i)}\|}.$$

So (9.33) becomes

$$\max_v f(v) \geq \max_{s_i=\pm 1} \frac{\sum_{i=1}^{p} f(v^{-(i)})}{\left\|\sum_{i=1}^{p} s_i v^{-(i)}\right\|} = \frac{\sum_{i=1}^{p} f(v^{-(i)})}{\min_{s_i=\pm 1}\left\|\sum_{i=1}^{p} s_i v^{-(i)}\right\|}. \tag{9.34}$$

Based on (9.32) and (9.34), we can get

$$\frac{\min\limits_{v} f(v)}{\max\limits_{v} f(v)} \leq \frac{\frac{1}{p}\sum\limits_{i=1}^{p} f(v^{-(i)})}{\frac{\sum\limits_{i=1}^{p} f(v^{-(i)})}{\min\limits_{s_i=\pm 1}\left\|\sum\limits_{i=1}^{p} s_i v^{-(i)}\right\|}} = \frac{1}{p}\min_{s_i=\pm 1}\left\|\sum_{i=1}^{p} s_i v^{-(i)}\right\|.$$

So we have

$$\max_{u_1,\ldots u_p}\frac{\min\limits_{v} f(v)}{\max\limits_{v} f(v)} \leq \frac{1}{p}\max_{u_1,\ldots u_p}\min_{s_i=\pm 1}\left\|\sum_{i=1}^{p} s_i v^{-(i)}\right\|. \tag{9.35}$$

Since solving the problem

$$\max_{u_1,\ldots u_p}\min_{s_i=\pm 1}\left\|\sum_{i=1}^{p} s_i v^{-(i)}\right\|$$

is equivalent to solving

$$\max_{u_1,\ldots u_p}\min_{s_i=\pm 1}\left\|\sum_{i=1}^{p} s_i v^{-(i)}\right\|^2,$$

we will try to solve the latter one in the following. We have

$$\max_{u_1,\ldots u_p}\min_{s_i=\pm 1}\left\|\sum_{i=1}^{p} s_i v^{-(i)}\right\|^2 = \max_{u_1,\ldots u_p}\min_{s_i=\pm 1} s^T\Sigma s,$$

where we have $\Sigma \in \mathbb{R}^{p\times p}$ and

$$\Sigma = \begin{pmatrix} 1 & \left(v^{-(1)}\right)^T v^{-(2)} & \left(v^{-(1)}\right)^T v^{-(3)} & \cdots & \left(v^{-(1)}\right)^T v^{-(p)} \\ \left(v^{-(2)}\right)^T v^{-(1)} & 1 & \left(v^{-(2)}\right)^T v^{-(3)} & \cdots & \left(v^{-(2)}\right)^T v^{-(p)} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \left(v^{-(p)}\right)^T v^{-(1)} & \left(v^{-(p)}\right)^T v^{-(2)} & \left(v^{-(p)}\right)^T v^{-(3)} & \cdots & 1 \end{pmatrix}.$$

We claim that $\min\limits_{s_i=\pm 1} s^T\Sigma s$ is upper bounded by p, and $\min\limits_{s_i=\pm 1} s^T\Sigma s = p$ when

$$\left(v^{-(i)}\right)^T v^{-(j)} = 0, \forall i \neq j.$$

We can see that if there are some i, j $(i \neq j)$, such that $\left(v^{-(i)}\right)^T v^{-(j)} \neq 0$, then there

exists some s, such that $s^T\Sigma s \leq p$. Suppose there does not exist such s, which means for any s, the following holds,

$$s^T\Sigma s > p. \tag{9.36}$$

Since we have

$$\begin{aligned}\sum_{s_i=\pm 1} s^T\Sigma s &= \sum_{s\in\{s:s_k=\pm 1\}}\sum_{i,j} s_i s_j \Sigma_{ij} = \sum_{s\in\{s:s_k=\pm 1\}}\left(p + \sum_{i\neq j} s_i s_j \Sigma_{ij}\right) \\ &= 2^p p + \sum_{s\in\{s:s_k=\pm 1\}}\sum_{i\neq j} s_i s_j \Sigma_{ij} = 2^p p,\end{aligned}$$

this will lead to $\sum_{s_i=\pm 1} s^T\Sigma s > 2^p p$, which is a contradiction of (9.36). So we proved that our claim is true, which says

$$\min_{s_i=\pm 1} s^T\Sigma s \leq p,$$

and when $\left(v^{-(i)}\right)^T v^{-(j)} = 0, \forall i \neq j$, which means $\Sigma = I_p$, we have $\min_{s_i=\pm 1} s^T\Sigma s = p$.

We know that $v^{-(i)}$'s only depends on u_i's, and when $u_i^T u_j = 0, \forall i \neq j$, we have $\left(v^{-(i)}\right)^T v^{-(j)} = 0, \forall i \neq j$. So when the following holds,

$$u_i^T u_j = 0, \forall i \neq j,$$

$\min_{s_i=\pm 1} s^T\Sigma s$ achieves the maximum value, which is p. Therefore we get

$$\max_{u_1,\ldots u_p}\min_{s_i=\pm 1}\left\|\sum_{i=1}^{p} s_i v^{-(i)}\right\|^2 = \max_{u_1,\ldots u_p}\min_{s_i=\pm 1} s^T\Sigma s = p,$$

which leads to

$$\max_{u_1,\ldots u_p}\min_{s_i=\pm 1}\left\|\sum_{i=1}^{p} s_i v^{-(i)}\right\| = \sqrt{p}. \tag{9.37}$$

Based on (9.35) and (9.37), we have

$$\max_{u_1,\dots u_p} \frac{\min_v f(v)}{\max_v f(v)} \leq \frac{\sqrt{p}}{p}. \tag{9.38}$$

Next if we can prove that when $u_i^T u_j = 0, \forall i \neq j$, the following holds, $\frac{\min_v f(v)}{\max_v f(v)} = \frac{\sqrt{p}}{p}$; combined with (9.38), we can arrive at the conclusion and finish the proof of the Lemma.

Let us assume

$$u_i^T u_j = 0, \forall i \neq j.$$

Without loss of generality, we can assume $u_i = e_i, \forall i \neq j$, where e_i's are the basic vectors of $\mathbb{R}^p$. Then the following holds,

$$f(v) = \sum_{i=1}^{p} |v_i|, v \in \mathbb{S}^{p-1}.$$

We can easily verify the following, $\min_v f(v) = 1$, and $\max_v f(v) = \sqrt{p}$. So when $u_i^T u_j = 0, \forall i \neq j$, we have

$$\frac{\min_v f(v)}{\max_v f(v)} = \frac{\sqrt{p}}{p}.$$

Combined what we get from (9.38), that is, $\frac{\sqrt{p}}{p}$ is the upper bound of $\max_{u_1,\dots u_p} \frac{\min_v f(v)}{\max_v f(v)}$, we finished the proof. □ □

B.10 Proof of Lemma 2.4.1

Proof. As we have

$$\min_{x:\|x\|=1,\langle x,v\rangle=\theta} \|x+B\|^2 = \min_{x:\|x\|=1,\langle x,v\rangle=\theta} 1 + \|B\|^2 + 2\langle x, B\rangle,$$

the problem (4.17) is equivalent to

$$\min_{x:\|x\|=1,\langle x,v\rangle=\theta} \langle x, B\rangle . \tag{10.39}$$

Suppose x^* is the solution to the above problem (4.17). Then x^* is the farthest point to B on the circle that satisfies the constraints $\|x\| = 1, \langle x, v\rangle = \theta$. The three points $x^*, v,$ and B must be on a same plane. Therefore, we can assume

$$x^* = av + bB. \tag{10.40}$$

Bringing (10.40) into (10.39), we have

$$\min_{x:\|x\|=1,\langle x,v\rangle=\theta} \langle x, B\rangle = \min_{a,b:\|av+bB\|=1,\langle av+bB,v\rangle=\theta} \langle av + bB, B\rangle ,$$

which is equivalent to

$$\min_{a,b} \quad av^T B + bB^T B \tag{10.41}$$

$$\text{s.t.} \quad \begin{cases} a^2 + b^2\|B\|^2 + 2abv^T B & = 1 \\ a + bv^T B & = \cos\theta. \end{cases} \tag{10.42}$$

Bringing the second equation in the constraints (10.42), that is,

$$a = \cos\theta - bv^T B \tag{10.43}$$

into (10.41), we have

$$\begin{aligned} \min_{b} \quad & v^T B\cos\theta + b\left(B^T B - (v^T B)^2\right) \\ \text{s.t.} \quad & b^2\left(B^T B - (v^T B)^2\right) + \cos\theta^2 = 1. \end{aligned} \tag{10.44}$$

Then the solution to (10.44) is

$$b = \pm \frac{\sin\theta}{\sqrt{B^T B - (v^T B)^2}}.$$

Since $B^T B - (v^T B)^2 \geq 0$, the minimum is achieved when

$$b = -\frac{|\sin\theta|}{\sqrt{B^T B - (v^T B)^2}}. \tag{10.45}$$

Combining (10.45) with (10.43), we can get the solution. □ □

B.11 Proof of Theorem 2.4.2

Proof. If $\theta \in [0, \pi)$, the square of the denominator of (4.18) becomes

$$1 + 2v^T B\cos\theta + B^T B - 2\sin\theta\sqrt{B^T B - (v^T B)^2} = 1 + B^T B + 2\sqrt{B^T B}\sin(\alpha - \theta),$$

where

$$\begin{aligned} \sin\alpha &= \frac{v^T B}{\sqrt{B^T B}}, \\ \cos\alpha &= \frac{\sqrt{B^T B - (v^T B)^2}}{\sqrt{B^T B}}. \end{aligned}$$

Similarly, if $\theta \in [\pi, 2\pi)$, then the square of the denominator of (4.18) becomes

$$1 + 2v^T B\cos\theta + B^T B + 2\sin\theta\sqrt{B^T B - (v^T B)^2} = 1 + B^T B + 2\sqrt{B^T B}\sin(\alpha + \theta),$$

where α is the same defined as above.

Hence, for $\theta \in [0, \pi)$, we have

$$f(\theta) = \frac{|\cos\theta| + A}{\sqrt{1 + B^T B + 2\sqrt{B^T B}\sin(\alpha - \theta)}};$$

for $\theta \in [\pi, 2\pi)$, we have

$$f(\theta) = \frac{|\cos\theta| + A}{\sqrt{1 + B^T B + 2\sqrt{B^T B}\sin(\alpha + \theta)}},$$

which is equivalent to

$$f(\theta) = \frac{|\cos\theta| + A}{\sqrt{1 + B^T B + 2\sqrt{B^T B}\sin(\alpha + \theta)}},$$

where $\theta \in [-\pi, 0)$, which is also equivalent to

$$f(\theta) = \frac{|\cos\theta| + A}{\sqrt{1 + B^T B + 2\sqrt{B^T B}\sin(\alpha - \theta)}},$$

where $\theta \in [0, \pi)$.

So the problem we want to solve is actually to maximize

$$f(\theta) = \frac{|\cos\theta| + A}{\sqrt{1 + B^T B + 2\sqrt{B^T B}\sin(\alpha - \theta)}}$$

on $\theta \in [0, \pi)$.

Under the first order condition, we have that if θ^* maximizes $f(\theta)$, then $0 = f'(\theta^*)$.

When $\theta \in [0, \frac{\pi}{2})$, the first order differentiable function of $f(\theta)$ can be written as

$$f'(\theta) = \frac{-(1 + B^T B)\sin\theta + \sqrt{B^T B}\,[\cos\alpha + A\cos(\alpha - \theta) - \sin\theta\sin(\alpha - \theta)]}{\Big(1 + B^T B + 2\sqrt{B^T B}\sin(\alpha - \theta)\Big)^{3/2}};$$

When $\theta \in [\frac{\pi}{2}, \pi)$, the first order differentiable function of $f(\theta)$ can be written as

$$f'(\theta) = \frac{(1 + B^T B)\sin\theta + \sqrt{B^T B}\,[-\cos\alpha + A\cos(\alpha - \theta) + \sin\theta\sin(\alpha - \theta)]}{\Big(1 + B^T B + 2\sqrt{B^T B}\sin(\alpha - \theta)\Big)^{3/2}}.$$

If we define function $g(\theta)$ as the following

$$g(\theta) = \begin{cases} \sqrt{B^T B}\,[\cos\alpha + A\cos(\alpha-\theta) - \sin\theta\sin(\alpha-\theta)] & \\ -(1+B^T B)\sin\theta, & \text{if } \theta \in [0, \frac{\pi}{2}), \\ \sqrt{B^T B}\,[-\cos\alpha + A\cos(\alpha-\theta) + \sin\theta\sin(\alpha-\theta)] & \\ +(1+B^T B)\sin\theta & \text{if } \theta \in [\frac{\pi}{2}, \pi), \end{cases}$$

Then our goal becomes to find the zeros of the function $g(\theta)$. □ □

APPENDIX C

PROOFS IN CHAPTER 3

C.1 Proof of Proposition 3.3.1

Proof. If we denote $\boldsymbol{v}$ as $\boldsymbol{v} = \boldsymbol{a} + i\boldsymbol{b}$, where $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^N$. Then matrix $\boldsymbol{X}$ can be expressed as

$$\boldsymbol{X} = \boldsymbol{v}\boldsymbol{v}^* = (\boldsymbol{a} + i\boldsymbol{b})(\boldsymbol{a} + i\boldsymbol{b})^* = (\boldsymbol{a}\boldsymbol{a}' - \boldsymbol{b}\boldsymbol{b}') + i(\boldsymbol{a}\boldsymbol{b}' + \boldsymbol{b}\boldsymbol{a}').$$

According to the definition of matrix $\boldsymbol{Z}$ in (3.6), we have

$$\boldsymbol{Z} = \begin{bmatrix} \boldsymbol{a}\boldsymbol{a}' - \boldsymbol{b}\boldsymbol{b}' & -(\boldsymbol{a}\boldsymbol{b}' + \boldsymbol{b}\boldsymbol{a}') \\ (\boldsymbol{a}\boldsymbol{b}' + \boldsymbol{b}\boldsymbol{a}') & \boldsymbol{a}\boldsymbol{a}' - \boldsymbol{b}\boldsymbol{b}' \end{bmatrix},$$

which can also be written as

$$\begin{bmatrix} \boldsymbol{a} & \boldsymbol{b} \\ \boldsymbol{b} & -\boldsymbol{a} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{a}' & -\boldsymbol{b}' \\ -\boldsymbol{b}' & -\boldsymbol{a}' \end{bmatrix}.$$

Therefore, the rank of matrix $\boldsymbol{Z}$ satisfies rank$\boldsymbol{Z} \leq 2$.

Next we show that rank of $\boldsymbol{Z}$ cannot be smaller than 2. Recall that rank of matrix $\boldsymbol{X}$ is 1, which is equivalent to the statement that there is only one nonzero solution to the equation

$$\boldsymbol{X}\boldsymbol{x} = \lambda\boldsymbol{x}. \tag{1.1}$$

(Here the vectors that have different lengths but the same or opposite directions are not seen as different solutions.) Assume $\boldsymbol{x} = \boldsymbol{\alpha} + i\boldsymbol{\beta}$. Thenonly one nonzero solution existing for equation (1.1) is equivalent to saying that there is only one solution regarding $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$

to the equation

$$(\mathcal{R}\boldsymbol{X}\boldsymbol{\alpha} - \mathcal{I}\boldsymbol{X}\boldsymbol{\beta}) + i(\mathcal{R}\boldsymbol{X}\boldsymbol{\beta} + \mathcal{I}\boldsymbol{X}\boldsymbol{\alpha}) = \lambda\boldsymbol{\alpha} + i\lambda\boldsymbol{\beta},$$

that is, there is only one solution regarding $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ to the equations

$$\mathcal{R}\boldsymbol{X}\boldsymbol{\alpha} - \mathcal{I}\boldsymbol{X}\boldsymbol{\beta} = \lambda\boldsymbol{\alpha}, \quad \mathcal{I}\boldsymbol{X}\boldsymbol{\alpha} + \mathcal{R}\boldsymbol{X}\boldsymbol{\beta} = \lambda\boldsymbol{\beta}, \tag{1.2}$$

which is equivalent to saying that there is only one nonzero solution of λ to the equation

$$\begin{bmatrix} \mathcal{R}\boldsymbol{X}\boldsymbol{\alpha} - \mathcal{I}\boldsymbol{X}\boldsymbol{\beta} \\ \mathcal{I}\boldsymbol{X}\boldsymbol{\alpha} + \mathcal{R}\boldsymbol{X}\boldsymbol{\beta} \end{bmatrix} = \begin{bmatrix} \lambda\boldsymbol{\alpha} \\ \lambda\boldsymbol{\beta} \end{bmatrix}.$$

If we denote $\boldsymbol{y} \in \mathbb{R}^{2N}$, then the above equation can be rewritten as $\boldsymbol{Z}\boldsymbol{y} = \lambda\boldsymbol{y}$, which means matrix $\boldsymbol{Z}$ has at least one nonzero eigenvalue. For that eigenvalue, the corresponding eigenvectors are at least $[\alpha', \beta']'$ and $[-\beta', \alpha']'$. So the rank of matrix $\boldsymbol{Z}$ is at least 2.

Therefore, the rank of matrix $\boldsymbol{Z}$ is equal to 2. Thus proof is completed. □

C.2 Proof of Theorem 3.3.2

Proof. Before the proof, we need the following theorem from [75]:

Theorem C.2.1. *(Theorem 3.4 in [75]) Let $\boldsymbol{A} \in \mathbb{S}^n$ have eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots \lambda_n$. Then, we have*

$$\max_{\boldsymbol{U} \in \Phi_{n,\mathcal{K}}} < \boldsymbol{A}, \boldsymbol{U} >= \sum_{i=1}^{\mathcal{K}} \lambda_i,$$

where $\boldsymbol{U} \in \Phi_{n,\mathcal{K}} = \{\boldsymbol{U} \in \mathbb{S}^n : 0 \preceq \boldsymbol{U} \preceq I, tr\boldsymbol{U} = \mathcal{K}\}$.

By letting $n = 2N$, $\mathcal{K} = 1$, $\boldsymbol{A} = \boldsymbol{Z}$, and $\boldsymbol{U} = \boldsymbol{I} - \boldsymbol{W}$ in Theorem C.2.1, we have

$$\max_{\boldsymbol{W} \in \Phi} < \boldsymbol{Z}, \boldsymbol{I} - \boldsymbol{W} >= \lambda_1(\boldsymbol{Z}) + \lambda_2(\boldsymbol{Z}). \tag{2.3}$$

As we know

$$\lambda_1(\boldsymbol{Z}) + \lambda_2(\boldsymbol{Z}) = \text{tr}(\boldsymbol{Z}) - \sum_{i=3}^{2N} \lambda_i(\boldsymbol{Z}),$$

we can get

$$\begin{aligned}
\sum_{i=3}^{2N} \lambda_i(\boldsymbol{Z}) &=< \boldsymbol{Z}, \boldsymbol{I} > - \max_{\boldsymbol{W} \in \Phi} < \boldsymbol{Z}, \boldsymbol{I} - \boldsymbol{W} > \\
&=< \boldsymbol{Z}, \boldsymbol{I} > + \min_{\boldsymbol{W} \in \Phi} < \boldsymbol{Z}, \boldsymbol{W} - \boldsymbol{I} > \\
&= \min_{\boldsymbol{W} \in \Phi} < \boldsymbol{Z}, \boldsymbol{W} > .
\end{aligned}$$

□

C.3 Proof of Theorem 3.3.3

Proof. From Theorem 3.3.2 we can get that

$$\sum_{i=3}^{2N} \lambda_i(\boldsymbol{Z}) \leq \min_{\boldsymbol{W} \in \Phi} \text{tr}(\boldsymbol{W}\boldsymbol{Z}).$$

Since $\boldsymbol{Z}$ satisfies $\boldsymbol{Z} \succeq 0$, we have $\sum_{i=3}^{2N} \lambda_i(\boldsymbol{Z}) \geq 0$. Therefore, equation (3.11) is a sufficient condition for rank$\boldsymbol{Z} = 2$. The necessity can also be confirmed by Theorem 3.3.2. □

C.4 Proof of Theorem 3.5.1

Proof. In each iteration k, the matrix $\boldsymbol{W}^{(k)}$ and $\boldsymbol{Z}^{(k)}$ are positive definite matrices. We first prove that tr$(\boldsymbol{W}^{(k)}\boldsymbol{Z}^{(k)})$ is lower bounded.

According to the properties of positive semidefinite matrices, we can say that there exists some matrix $\boldsymbol{A}$, such that $\boldsymbol{Z}^{(k)} = \boldsymbol{A}^{1/2}\boldsymbol{A}^{1/2}$. Then, tr$(\boldsymbol{W}^{(k)}\boldsymbol{Z}^{(k)})$ can be written as

$$\text{tr}(\boldsymbol{W}^{(k)}\boldsymbol{Z}^{(k)}) = \text{tr}(\boldsymbol{W}^{(k)}\boldsymbol{A}^{1/2}\boldsymbol{A}^{1/2}) = \text{tr}(\boldsymbol{A}^{1/2}\boldsymbol{W}^{(k)}\boldsymbol{A}^{1/2}). \tag{4.4}$$

As for any vector x, we have

$$x^T(\boldsymbol{A}^{1/2}\boldsymbol{W}^{(k)}\boldsymbol{A}^{1/2})x = (\boldsymbol{A}^{1/2}x)^T\boldsymbol{W}^{(k)}(\boldsymbol{A}^{1/2}x),$$

whose trace is positive as $\boldsymbol{W}^{(k)}$ is positive semidefinite. Therefore, $\boldsymbol{A}^{1/2}\boldsymbol{W}^{(k)}\boldsymbol{A}^{1/2}$ is a positive semidefinite matrix, which indicates

$$\text{tr}(\boldsymbol{A}^{1/2}\boldsymbol{W}^{(k)}\boldsymbol{A}^{1/2}) \geq 0.$$

As the equality in (4.4), we have $\text{tr}(\boldsymbol{W}^{(k)}\boldsymbol{Z}^{(k)}) \geq 0$. So the series $\left\{\text{tr}(\boldsymbol{W}^{(k)}\boldsymbol{Z}^{(k)})\right\}$ can be lower bounded by 0.

Next, because the two convex optimization problems in each iteration are all minimizing problems, the following observation can be obtained:

$$\text{tr}(\boldsymbol{W}^{(k)}\boldsymbol{Z}^{(k)}) \leq \text{tr}(\boldsymbol{W}^{(k-1)}\boldsymbol{Z}^{(k)}) \leq \text{tr}(\boldsymbol{W}^{(k-1)}\boldsymbol{Z}^{(k-1)}).$$

Therefore the series $\left\{\text{tr}(\boldsymbol{W}^{(k)}\boldsymbol{Z}^{(k)})\right\}$ is a decreasing series, and it has a upper bound $\text{tr}(\boldsymbol{W}^{(0)}\boldsymbol{Z}^{(0)})$.

Since the the series $\left\{\text{tr}(\boldsymbol{W}^{(k)}\boldsymbol{Z}^{(k)})\right\}$ is both bounded and decreasing, it is convergent.

□

C.5 Proof of Theorem 3.5.3

Proof. Since the rank of matrix $\boldsymbol{Z}$ is 2, we can write $\boldsymbol{Z}$ as

$$\boldsymbol{Z} = \boldsymbol{v}\boldsymbol{v}^T,$$

where $\boldsymbol{v}$ is the $2N \times 2$ matrix. Therefore, the condition (3.11) can be written as

$$\text{tr}(\boldsymbol{W}\boldsymbol{v}\boldsymbol{v}^T) = 0. \tag{5.5}$$

As $\boldsymbol{W}$ is a nonnegative semidefinite symmetric matrix, it can be decomposed as

$$\boldsymbol{W} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T, \tag{5.6}$$

where columns of $\boldsymbol{U}$ is the eigenvectors, $\boldsymbol{\Lambda}$ is a diagonal matrix whose diagonal elements are eigenvalues. Then, by substituting (5.6) into (5.5), we can get

$$\text{tr}(\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T\boldsymbol{v}\boldsymbol{v}^T) = 0.$$

Furthermore, we have

$$\text{tr}\left(\boldsymbol{\Lambda}(\boldsymbol{U}^T\boldsymbol{v})(\boldsymbol{U}^T\boldsymbol{v})^T\right) = 0. \tag{5.7}$$

Assume the eigenvalues of $\boldsymbol{W}$ are $\{\lambda_1, \ldots, \lambda_{2N}\}$. Then, matrix $\boldsymbol{\Lambda}$ is equal to

$$\boldsymbol{\Lambda} = \text{diag}\left\{\lambda_1, \ldots, \lambda_{2N}\right\}.$$

According to condition (3.9), we know

$$\boldsymbol{0} \preceq \boldsymbol{W} \preceq \boldsymbol{I},$$

which implies

$$\boldsymbol{0} \preceq \boldsymbol{\Lambda} = \text{diag}\left\{\lambda_1, \ldots, \lambda_{2N}\right\} \preceq \boldsymbol{I}.$$

Therefore, we have

$$0 \leq \lambda_i \leq 1,\ i = 1, \ldots, 2N. \tag{5.8}$$

Condition (3.9) also tells us

$$\text{tr}(\boldsymbol{W}) = 2N - 2,$$

which implies

$$\sum_{i=1}^{2N} \lambda_i = 2N - 2. \tag{5.9}$$

In (5.7), if we let $\boldsymbol{w} = \boldsymbol{U}^T\boldsymbol{v}$, then we have

$$\text{tr}(\boldsymbol{\Lambda}\boldsymbol{w}\boldsymbol{w}^T) = 0,$$

which can also be written as

$$\sum_{i=1}^{2N} \lambda_i(w_{i1}^2 + w_{i2}^2) = 0.$$

The above equation tells us that if $w_{ij} \neq 0$, then we must have $\lambda_i = 0$. From (5.8) and (5.9), we know the λ_i's satisfy that there exist $1 \leq i_0 \leq 2N$, such that $\lambda_{i_0} = 0$, and $\lambda_i = 1,$ for $i \neq i_0$. So matrix $\boldsymbol{\Lambda}$ can be constructed. Therefore, the matrix $\boldsymbol{w}$ must satisfy $w_{i1} = w_{i2} = 0$, for $i \neq i_0$. Assume $\boldsymbol{U} = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{2N})$. Then, because of $\boldsymbol{w} = \boldsymbol{U}^T\boldsymbol{v}$, we have $\boldsymbol{u}_i^T\boldsymbol{v} = [w_{i1}, w_{i2}]$. Therefore, we know that for $i \neq i_0$, $\boldsymbol{u}_i^T\boldsymbol{v} = \boldsymbol{0}$. So matrix $\boldsymbol{U}$ can also be constructed. Overall, matrix $\boldsymbol{W}$ exists and can be constructed. □

C.6 Convergence Analysis for State Estimation

Similar to the case of power flow analysis, we have Theorem 3.5.2 to guarantee the local optimality. The proofs are as follows:

C.6.1 Proof of Theorem 3.5.2

Proof. Based on the same proof in the proof of Theorem 3.5.1, we can show that $\text{tr}(\boldsymbol{W}^{(k)}\boldsymbol{Z}^{(k)})$ is lower bounded by 0, that is,

$$\text{tr}(\boldsymbol{W}^{(k)}\boldsymbol{Z}^{(k)}) \geq 0.$$

As the other term $\sum_{j=1}^{M} |r_j^{(k)}|$ is also lower bounded by 0, we get the series

$$\left\{ \sum_{j=1}^{M} |r_j^{(k)}| + \text{tr}(\boldsymbol{W}^{(k)} \boldsymbol{Z}^{(k)}) \right\}$$

is also lower bounded by 0. Before we proceed to the next step, the equivalent representation of the series is stated as follows:

$$\sum_{j=1}^{M} |r_j^{(k)}| + \text{tr}(\boldsymbol{W}^{(k)} \boldsymbol{Z}^{(k)}) = \sum_{j=1}^{M} |z_j - \text{tr}\left(\boldsymbol{M}_j \boldsymbol{X}^{(k)}\right)| + \text{tr}(\boldsymbol{W}^{(k)} \boldsymbol{Z}^{(k)}).$$

Next, same as in the proof of Theorem 3.5.1, because the two convex optimization problems in each iteration are all minimizing problems, the following observation can be obtained:

$$\begin{aligned} \sum_{j=1}^{M} |z_j - \text{tr}\left(\boldsymbol{M}_j \boldsymbol{X}^{(k)}\right)| + \text{tr}(\boldsymbol{W}^{(k)} \boldsymbol{Z}^{(k)}) &\leq \sum_{j=1}^{M} |z_j - \text{tr}\left(\boldsymbol{M}_j \boldsymbol{X}^{(k)}\right)| + \text{tr}(\boldsymbol{W}^{(k-1)} \boldsymbol{Z}^{(k)}) \\ &\leq \sum_{j=1}^{M} |z_j - \text{tr}\left(\boldsymbol{M}_j \boldsymbol{X}^{(k-1)}\right)| + \text{tr}(\boldsymbol{W}^{(k-1)} \boldsymbol{Z}^{(k-1)}). \end{aligned}$$

Therefore the series $\left\{ \sum_{j=1}^{M} |r_j^{(k)}| + \text{tr}(\boldsymbol{W}^{(k)} \boldsymbol{Z}^{(k)}) \right\}$ is a decreasing series, and it has a upper bound $\sum_{j=1}^{M} |r_j^{(0)}| + \text{tr}(\boldsymbol{W}^{(0)} \boldsymbol{Z}^{(0)})$.

Since the the series $\left\{ \sum_{j=1}^{M} |r_j^{(k)}| + \text{tr}(\boldsymbol{W}^{(k)} \boldsymbol{Z}^{(k)}) \right\}$ is both bounded and decreasing, it is convergent. □

REFERENCES

[1] K. Pearson, "On lines and planes of closest fit to systems of points in space," *Philosophical Magazine*, vol. 2, pp. 559–572, 6 1901.

[2] P. Comon, "Independent component analysis, a new concept?" *Signal processing*, vol. 36, no. 3, pp. 287–314, 1994.

[3] W. S. Torgerson, "Theory and methods of scaling.," 1958.

[4] H. Hotelling, "Relations between two sets of variates," *Biometrika*, vol. 28, no. 3/4, pp. 321–377, 1936.

[5] J. R. Kettenring, "Canonical analysis of several sets of variables," *Biometrika*, vol. 58, no. 3, pp. 433–451, 1971.

[6] G. Andrew, R. Arora, J. Bilmes, and K. Livescu, "Deep canonical correlation analysis," in *International Conference on Machine Learning*, 2013, pp. 1247–1255.

[7] P. L. Lai and C. Fyfe, "A neural implementation of canonical correlation analysis," *Neural Networks*, vol. 12, no. 10, pp. 1391–1397, 1999.

[8] ——, "Kernel and nonlinear canonical correlation analysis," *International Journal of Neural Systems*, vol. 10, no. 05, pp. 365–377, 2000.

[9] F. R. Bach and M. I. Jordan, "Kernel independent component analysis," *Journal of machine learning research*, vol. 3, no. Jul, pp. 1–48, 2002.

[10] B. Chang, U. Kruger, R. Kustra, and J. Zhang, "Canonical correlation analysis based on hilbert-schmidt independence criterion and centered kernel target alignment," in *International Conference on Machine Learning*, 2013, pp. 316–324.

[11] T. W. Anderson, "Estimating linear restrictions on regression coefficients for multivariate normal distributions," *The Annals of Mathematical Statistics*, pp. 327–351, 1951.

[12] A. J. Izenman, "Reduced-rank regression for the multivariate linear model," *Journal of multivariate analysis*, vol. 5, no. 2, pp. 248–264, 1975.

[13] R. D. Cook and C. M. Setodji, "A model-free test for reduced rank in multivariate regression," *Journal of the American Statistical Association*, vol. 98, no. 462, pp. 340–351, 2003.

[14] X. Yin, "Canonical correlation analysis based on information theory," *Journal of multivariate analysis*, vol. 91, no. 2, pp. 161–176, 2004.

[15] A. Mukherjee and J. Zhu, "Reduced rank ridge regression and its kernel extensions," *Statistical analysis and data mining: the ASA data science journal*, vol. 4, no. 6, pp. 612–622, 2011.

[16] K.-C. Li, "Sliced inverse regression for dimension reduction," *Journal of the American Statistical Association*, vol. 86, no. 414, pp. 316–327, 1991.

[17] R. D. Cook and L. Forzani, "Likelihood-based sufficient dimension reduction," *Journal of the American Statistical Association*, vol. 104, no. 485, pp. 197–208, 2009.

[18] K. Chen and Y. Ma, "Analysis of double single index models," *Scandinavian Journal of Statistics*, vol. 44, no. 1, pp. 1–20, 2017.

[19] R. Iaci, X. Yin, and L. Zhu, "The dual central subspaces in dimension reduction," *Journal of Multivariate Analysis*, vol. 145, pp. 178–189, 2016.

[20] G. J. Székely, M. L. Rizzo, N. K. Bakirov, *et al.*, "Measuring and testing dependence by correlation of distances," *The annals of statistics*, vol. 35, no. 6, pp. 2769–2794, 2007.

[21] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[22] W. Sheng and X. Yin, "Sufficient dimension reduction via distance covariance," *Journal of Computational and Graphical Statistics*, vol. 25, no. 1, pp. 91–104, 2016.

[23] ——, "Direction estimation in single-index models via distance covariance," *Journal of Multivariate Analysis*, vol. 122, pp. 148–161, 2013.

[24] P. D. Tao and L. T. H. An, "Convex analysis approach to dc programming: Theory, algorithms and applications," *Acta Mathematica Vietnamica*, vol. 22, no. 1, pp. 289–355, 1997.

[25] G. H. Golub and C. F. Van Loan, *Matrix computations*. JHU Press, 2012, vol. 3.

[26] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley interdisciplinary reviews: computational statistics*, vol. 2, no. 4, pp. 433–459, 2010.

[27] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent component analysis*. John Wiley & Sons, 2004, vol. 46.

[28] J. B. Kruskal and M. Wish, *Multidimensional scaling*. Sage, 1978, vol. 11.

[29] R. D. Cook, B. Li, and F. Chiaromonte, "Envelope models for parsimonious and efficient multivariate linear regression," *Statistica Sinica*, pp. 927–960, 2010.

[30] R. D. Cook and Z. Su, "Scaled envelopes: Scale-invariant and efficient estimation in multivariate linear regression," *Biometrika*, vol. 100, no. 4, pp. 939–954, 2013.

[31] Z. Su and R. D. Cook, "Partial envelopes for efficient estimation in multivariate linear regression," *Biometrika*, vol. 98, no. 1, pp. 133–146, 2011.

[32] ——, "Inner envelopes: Efficient estimation in multivariate linear regression," *Biometrika*, vol. 99, no. 3, pp. 687–702, 2012.

[33] K.-C. Li, Y. Aragon, K. Shedden, and C Thomas Agnan, "Dimension reduction for multivariate response data," *Journal of the American Statistical Association*, vol. 98, no. 461, pp. 99–109, 2003.

[34] R. D. Cook and S. Weisberg, "Comment," *Journal of the American Statistical Association*, vol. 86, no. 414, pp. 328–332, 1991.

[35] H. Wang and Y. Xia, "Sliced regression for dimension reduction," *Journal of the American Statistical Association*, vol. 103, no. 482, pp. 811–821, 2008.

[36] R. Shumway, A. Azari, and Y Pawitan, "Modeling mortality fluctuations in los angeles as functions of pollution and weather effects," *Environmental Research*, vol. 45, no. 2, pp. 224–241, 1988.

[37] B. Li, S. Wen, and L. Zhu, "On a projective resampling method for dimension reduction with multivariate responses," *Journal of the American Statistical Association*, vol. 103, no. 483, pp. 1177–1186, 2008.

[38] G. J. Székely and M. L. Rizzo, "Testing for equal distributions in high dimension," *InterStat*, vol. 5, pp. 1–6, 2004.

[39] S. Bradley, A. Hax, and T. Magnanti, "Applied mathematical programming," 1977.

[40] X. Huo and G. J. Székely, "Fast computing for distance covariance," *Technometrics*, vol. 58, no. 4, pp. 435–447, 2016.

[41] C. Huang and X. Huo, "An efficient and distribution-free two-sample test based on energy statistics and random projections," *arXiv preprint arXiv:1707.04602*, 2017.

[42] I. H. Sloan and R. S. Womersley, "Extremal systems of points and numerical integration on the sphere," *Advances in Computational Mathematics*, vol. 21, no. 1-2, pp. 107–125, 2004.

[43] K. Hesse, I. H. Sloan, and R. S. Womersley, "Numerical integration on the sphere," in *Handbook of Geomathematics*, Springer, 2010, pp. 1185–1219.

[44] J. Brauchart, E Saff, I Sloan, and R Womersley, "Qmc designs: Optimal order quasi monte carlo integration schemes on the sphere," *Mathematics of computation*, vol. 83, no. 290, pp. 2821–2851, 2014.

[45] S. J. Wright, "Coordinate descent algorithms," *Mathematical Programming*, vol. 151, no. 1, pp. 3–34, 2015.

[46] G. J. Székely and M. L. Rizzo, "Brownian distance covariance," *The annals of applied statistics*, pp. 1236–1265, 2009.

[47] R. Lyons *et al.*, "Distance covariance in metric spaces," *The Annals of Probability*, vol. 41, no. 5, pp. 3284–3305, 2013.

[48] V. S. Korolyuk and Y. V. Borovskich, *Theory of U-statistics*. Springer Science & Business Media, 2013, vol. 273.

[49] W. Hoeffding, "A class of statistics with asymptotically normal distribution," in *Breakthroughs in Statistics*, Springer, 1992, pp. 308–334.

[50] R. v. Mises, "On the asymptotic distribution of differentiable statistical functions," *The annals of mathematical statistics*, vol. 18, no. 3, pp. 309–348, 1947.

[51] Y. Nesterov, "Efficiency of coordinate descent methods on huge-scale optimization problems," *SIAM Journal on Optimization*, vol. 22, no. 2, pp. 341–362, 2012.

[52] A. Chaudhuri and W. Hu, "A fast algorithm for computing distance correlation," *arXiv preprint arXiv:1810.11332*, 2018.

[53] H. Niederreiter, *Random number generation and quasi-Monte Carlo methods*. Siam, 1992, vol. 63.

[54] S. Asmussen and P. W. Glynn, *Stochastic simulation: algorithms and analysis*. Springer Science & Business Media, 2007, vol. 57.

[55] W. J. Morokoff and R. E. Caflisch, "Quasi-monte carlo integration," *Journal of computational physics*, vol. 122, no. 2, pp. 218–230, 1995.

[56] D. Bienstock and A. Verma, "Strong np-hardness of ac power flows feasibility," *arXiv preprint arXiv:1512.07315*, 2015.

[57] J. J. Grainger and W. D. Stevenson, *Power system analysis*. McGraw-Hill, 1994.

[58] G. Andersson, "Modelling and analysis of electric power systems," *ETH Zurich*, pp. 5–6, 2008.

[59] B. Stott, J. Jardim, and O. Alsaç, "Dc power flow revisited," *IEEE Transactions on Power Systems*, vol. 24, no. 3, pp. 1290–1300, 2009.

[60] U. G. Knight, *Power Systems Engineering and Mathematics: International Series of Monographs in Electrical Engineering*. Elsevier, 2017, vol. 3.

[61] W. F. Tinney and C. E. Hart, "Power flow solution by newton's method," *IEEE Transactions on Power Apparatus and systems*, no. 11, pp. 1449–1460, 1967.

[62] B. Stott and O. Alsac, "Fast decoupled load flow," *IEEE transactions on power apparatus and systems*, no. 3, pp. 859–869, 1974.

[63] Y.-C. Wu, A. S. Debs, and R. E. Marsten, "A direct nonlinear predictor-corrector primal-dual interior point algorithm for optimal power flows," *IEEE Transactions on power systems*, vol. 9, no. 2, pp. 876–883, 1994.

[64] Y. Weng, Q. Li, R. Negi, and M. Ilić, "Semidefinite programming for power system state estimation," in *2012 IEEE Power and Energy Society General Meeting*, IEEE, 2012, pp. 1–8.

[65] H. Zhu and G. B. Giannakis, "Power system nonlinear state estimation using distributed semidefinite programming," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 6, pp. 1039–1050, 2014.

[66] S. Bhela, V. Kekatos, and S. Veeramachaneni, "Enhancing observability in distribution grids using smart meter data," *IEEE Transactions on Smart Grid*, vol. 9, no. 6, pp. 5953–5961, 2018.

[67] Y. Zhang, R. Madani, and J. Lavaei, "Conic relaxations for power system state estimation with line measurements," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 3, pp. 1193–1205, 2018.

[68] G. Wang, H. Zhu, G. B. Giannakis, and J. Sun, "Robust power system state estimation from rank-one measurements," *IEEE Transactions on Control of Network Systems*, 2019.

[69] A. Gomez-Exposito and A. Abur, *Power system state estimation: theory and implementation*. CRC press, 2004.

[70] M. Irving, R. Owen, and M. Sterling, "Power-system state estimation using linear programming," in *Proceedings of the Institution of Electrical Engineers*, IET, vol. 125, 1978, pp. 879–885.

[71] W. W. Kotiuga and M Vidyasagar, "Bad data rejection properties of weughted least absolute value techniques applied to static state estimation," *IEEE Transactions on Power Apparatus and Systems*, no. 4, pp. 844–853, 1982.

[72] I. Molybog and J. Lavaei, "On sampling complexity of the semidefinite affine rank feasibility problem," in *Thirty-Third AAAI Conference on Artificial Intelligence*, 2019.

[73] A. J. Wood, B. F. Wollenberg, *et al.*, *Power generation, operation, and control*. John Wiley & Sons, 2013.

[74] J. Dattorro, *Convex optimization & Euclidean distance geometry*. Lulu. com, 2010.

[75] M. L. Overton and R. S. Womersley, "Optimality conditions and duality theory for minimizing sums of the largest eigenvalues of symmetric matrices," *Mathematical Programming*, vol. 62, no. 1-3, pp. 321–357, 1993.

[76] R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. Thomas, "Matpower: Steady-state operations, planning, and analysis tools for power systems research and education," *IEEE Transactions on power systems*, vol. 26, no. 1, pp. 12–19, 2011.

[77] J. Lavaei and S. H. Low, "Zero duality gap in optimal power flow problem," *IEEE Transactions on Power Systems*, vol. 27, no. 1, pp. 92–107, 2011.

[78] X. Bai, H. Wei, K. Fujisawa, and Y. Wang, "Semidefinite programming for optimal power flow problems," *International Journal of Electrical Power & Energy Systems*, vol. 30, no. 6-7, pp. 383–392, 2008.

[79] R. Madani, M. Ashraphijuo, and J. Lavaei, "Promises of conic relaxation for contingency-constrained optimal power flow problem," *IEEE Transactions on Power Systems*, vol. 31, no. 2, pp. 1297–1307, 2015.